Spring 2015

Privacy-preserving social network analysis

Christine Marie Task
Purdue University

Graduate School Form 30
(Revised 08/14)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Christine Task

Entitled
PRIVACY-PRESERVING SOCIAL NETWORK ANALYSIS

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Chris Clifton

David Gleich

Jennifer Neville

Mikhail Atallah

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): Chris Clifton

Approved by: Sunil Prabhakar / William Gorman 04/15/2015

Head of the Department Graduate Program Date

PRIVACY-PRESERVING SOCIAL NETWORK ANALYSIS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Christine Task

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2015

Purdue University

West Lafayette, Indiana

I dedicate this to my indefatigable husband Mark, without whom I'd have been lost; I'd also like to sincerely thank my advisor, who was available, always, with the very best advice. As a kid I dreamt of playing with problems like a puzzle or a game, and with the invaluable support of my committee, courses, and advisor, I got to.

Special mention is due to the following faithful coffee houses: The Runcible Spoon, Greyhouse, Killer ESP, and Rappahannock. My earnest thanks also to Pete, Nick and Erin–dear friends with spare beds, flexible schedules, and kind company on the long road between DC and West Lafayette.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

Figure Page

ABSTRACT

Task, Christine Ph.D., Purdue University, May 2015. Privacy-preserving Social Network Analysis. Major Professor: Chris Clifton.

Data privacy in social networks is a growing concern that threatens to limit access to important information contained in these data structures. Analysis of the graph structure of social networks can provide valuable information for revenue generation and social science research, but unfortunately, ensuring this analysis does not violate individual privacy is difficult. Simply removing obvious identifiers from graphs or even releasing only aggregate results of analysis may not provide sufficient protection. Differential privacy is an alternative privacy model, popular in data-mining over tabular data, that uses noise to obscure individuals' contributions to aggregate results and offers a strong mathematical guarantee that individuals' presence in the data-set is hidden. Analyses that were previously vulnerable to identification of individuals and extraction of private data may be safely released under differential-privacy guarantees. However, existing adaptations of differential privacy to social network analysis are often complex and have considerable impact on the utility of the results, making it less likely that they will see widespread adoption in the social network analysis world. In fact, social scientists still often use the weakest form of privacy protection, simple anonymization, in their social network analysis publications, [1–6].

We review the existing work in graph-privatization, including the two existing standards for adapting differential privacy to network data. We then propose *contributor-privacy* and *partition-privacy*, novel standards for differential privacy over network data, and introduce simple, powerful private algorithms using these standards for common network analysis techniques that were infeasible to privatize under previous differential privacy standards. We also ensure that privatized social net-

work analysis does not violate the level of rigor required in social science research, by proposing a method of determining statistical significance for paired samples under differential privacy using the Wilcoxon Signed-Rank Test, which is appropriate for non-normally distributed data.

Finally, we return to formally consider the case where differential privacy is not applied to data. Naive, deterministic approaches to privacy protection, including anonymization and aggregation of data, are often used in real world practice. De-anonymization research demonstrates that some naive approaches to privacy are highly vulnerable to reidentification attacks, and none of these approaches offer the robust guarantee of differential privacy. However, we propose that these methods fall across a range of protection: Some are better than others. In cases where adding noise to data is especially problematic, or acceptance and adoption of differential privacy is especially slow, it is critical to have a formal understanding of the alternatives. We define *De Facto Privacy*, a metric for comparing the relative privacy protection provided by deterministic approaches.

1. INTRODUCTION

There is tremendous value in the set of relationships among individuals, which touches on many important areas of scientific inquiry. Social networks are powerful abstractions of this information, applying a classical graph data-structure representing individuals as nodes and their relationships as edges. Social network analysis can yield valuable insights into the behavior of populations. For example, understanding how well-connected a network is can aid in the development of a word-of-mouth marketing campaign: How quickly will word of a product spread? Similar analysis is useful in epidemiology (predicting spread of a disease through connections in a population), or in learning analytics (studying how students' interactions impact learning).

However, data about people and their relationships is potentially sensitive and must be treated with care to preserve privacy. Generally, social network graphs are anonymized before being made available for analysis. For example, Figure 1.1 depicts one of the earliest applications of social network analysis to anthropological research; the beginnings of a schism in a university karate club was detected before the group decided to split in two, through surveying members about their social interactions. The graph was published as a simply anonymized network with node labels indicating subsequent group membership and leadership roles [7].

Unfortunately, releasing anonymized graphs may lead to re-identification of individuals within the network and disclosure of confidential information, with serious consequences for those involved. In the karate club network, anyone possessing minimal background knowledge of the group would be able to assign names to the two leader nodes (nodes **1** and **34**). Other nodes are distinctive as well: we know from details in the publication that **2,3** and **33** are likely to be the teaching assistants in each group, and we know that **9** was the only student to associate closely with leader A but remain in the group with leader B. A small amount of familiarity with the

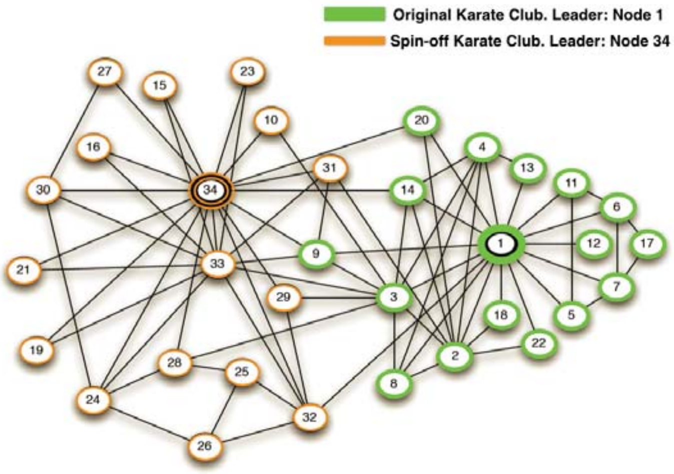

🟩 Original Karate Club. Leader: Node 1
🟧 Spin-off Karate Club. Leader: Node 34
27
15
23
10
20
16
4
13
31
11
30
34
14
6
1
12
17
9
21
33
29
3
7
18
5
19
22
28
2
25
8
24
32
26

Fig. 1.1.: An anonyzmized social network collected over the 34 members of a university karate club, shortly before a schism caused the group to split in two.

club would be sufficient to assign names to these nodes. Once one distinctive student has been identified in the graph, background knowledge about that student will help us uncover other students. For example, if we knew that *Alice* was the only student who split off after being friends with the leader of the original group, then we know *Alice* is node **32**. Say we know that *Bob* is friends with both the leader of the new group and TA **3** in the old group, and we've also heard that he has one partner he practices with privately. Then *Bob* must be **29** and *Alice*, node **32**, is his previously private partner. The process of mapping true names to anonymized individual data entries, in this case graph nodes, is known as 'de-anonymization'. In a de-anonymized data-set, sensitive information such as private relationship edges, node degrees, and node or edge properties is made fully public.

Fortunately, the karate club study was published in 1977; the club did not have a website, and the students were not on Twitter. This limited the public availability of background information that could be used to identify individuals in the network. To the best of our knowledge, the karate club network has not been de-anonymized.

By contrast, in 2007 Netflix released the 'Netflix Prize' data-set containing anonymized data about the viewing habits of its members, intended for public analysis by information retrieval researchers. Within a year, it had been demonstrated that wide-spread de-anonymization of individuals in the data-set was possible using publicly available background information from the Internet Movie Database [8]. By 2009, Netflix was involved in a lawsuit with one of its members who had been victimized by the resulting privacy invasion.

This object lesson about the dangers of publishing simply-anonymized data in the internet-era appears to not have been well-learned. In March of 2014, the Public Library of Science (PLOS), a major publisher of research in the biological and medical sciences, mandated that researchers make their data fully, publicly available as a condition for publication [9]. PLOS publications include researchers working on sensitive epidemiology social networks [10, 11]. Restrictions are permitted only in the case of sensitive human data, but anonymized data has in the past been considered to be sufficiently protected. In the learning analytics community, researchers still commonly include complete anonymized networks of students in their published papers, in a continuation of the practice applied in the 1977 karate club paper [1–6]. These networks may include sensitive node data such as course grades, discussion of academic difficulties, and private relationships between students. Meanwhile, public outcry about the privacy risks of educational data-analysis in K-12 schools threatens to entirely prevent the adoption of these techniques, despite evidence that they have potential to significantly improve students chances of success [12–14].

My work directly addresses the problem of creating effective, usable privatized social network analysis tools.

1.1 Challenges in Graph Privatization

Satisfying this need is non-trivial.

The strength of de-anonymization techniques: Privacy researchers have attempted to improve the security provided by graph anonymization techniques by adding noise to the node parameters and structure of the graph [15]. However, even a noisy graph structure with no node parameters whatsoever can be subject to deanonymization, particularly if an attacker has background knowledge of the network data [16,17]. Knowing the friendship relationships of a few individuals can make them identifiable in the released graph, leading to identification of their friends (and disclosure of information, such as other relationships, that those friends might not want publicly revealed). As global social networks become more broadly accessible, sources of extensive background knowledge are increasingly available [16].

The complexities of Differential Privacy: *Differential privacy* is a privacy standard developed for use on tabular data that provides strong guarantees of privacy without making assumptions about an attacker's background knowledge [18]. Differentially private queries inject randomized noise into query results to hide the impact of adding or removing an arbitrary individual from the data-set. Thus, an attacker with an arbitrarily high level of background knowledge cannot, with a high degree of probability, glean any new knowledge about individuals from differentially privatized results; in fact, the attacker cannot guess whether any given individual is present in the data at all.

While many of the privacy concerns associated with social-network analysis could be relieved by applying differential privacy guarantees to common social network analysis techniques, researchers have struggled to develop suitable adaptations of these techniques.

Two principal difficulties arise: The adaptation of differential privacy from tabular data to network data, and the high sensitivity of social-network metrics to relatively small changes in the network structure. This high sensitivity requires many existing differentially private network analysis techniques to add impractically large amounts of noise when operated on real world data, significantly affecting the utility of the privatized results [19].

The difficulties in real world adoption: Privacy protection is not a hypothetical concern for individuals appearing in social science and data-mining data-sets. Privatization techniques that are very mathematically or algorithmically complex may be less likely to be adopted by the members of the social science community who are collecting and studying human data [19–21]. Although this research may provide insightful theoretical solutions to the privacy problems they address, in the real world, those privacy problems remain.

1.2 Problem Statement

In this work, we will consider the question: "Is practically usable, privacy-preserving social network analysis feasible?" We will address this question in three ways:

- We will propose and demonstrate network privatization techniques that adapt differential privacy to satisfy these requirements for many social network analysis applications, including the computation of statistical significance on paired samples under differential privacy.
- We will also demonstrate that some types of analysis are inherently difficult to privatize.
- We will finally propose a metric for formally comparing the relative privacy protection provided by deterministic approaches, such as anonymity and aggregation, which have the advantage of not introducing noise to the data, but which fall short of satisfying the robust guarantee of differential privacy and are more vulnerable to attack.

A fundamental contribution of this dissertation is the following observation: In privacy-preserving contexts, it is vital to consider the intended application of the sensitive data and to use a data structure which requires no more complexity or detail than absolutely required to successfully achieve the intended objective. In social network applications the objective is to learn about the population abstracted by the network, and the network data-structure itself can be an optional intermediate step in the analysis process. Protecting privacy in a single complete, cohesive network is very challenging: building the network requires tying together information from thousands of individuals into a tightly interlinked whole in which any sufficiently pathological small change might dramatically affect global analysis results, and privatization essentially involves very carefully destroying that same network.

By introducing analyses that do not require a single cohesive network, but rather consider distributions of unlinked ego-network or sub-graph data, we can offer comparable strengths of privacy protection without going through the expensive steps of first constructing and then privatizing the complete social network graph. We demonstrate that it is possible to preserve, with high accuracy, the aggregate social patterns we actually seek to study while using sets of unlinked, relatively insensitive and naturally privacy-preserving data.

1.3 Contributions

Our first set of contributions involve identifying effective methods of adapting differential privacy to network data. Previous to our work, two models for applying differential privacy to social networks have arisen. *Node-privacy* limits the ability of an attacker to learn any information about an individual, but at a high cost in added noise. *Edge-privacy* protects against learning any particular relationship, but does not prevent learning information about an individual. This work introduces *contributor-privacy* and *partition-privacy*, models for differential privacy that can provide stronger protection for individuals than edge privacy while allowing important types of analysis that are not feasible under node privacy.

Additionally, we propose *De Facto Privacy* as a metric for objectively comparing the privacy provided by deterministic approaches, such as simple anonymity and aggregation, that fall short of satisfying differential privacy.

Specifically, this work provides:

1. A straightforward introduction to traditional differential privacy and the basics of social network analysis (Chapter 2);

2. A discussion of existing work in graph-privatization, including work on anonymity, de-anonymization, and the two existing differential-privacy standards for network data (Chapter 3).

3. The contribution of two new standards, *contributor-privacy* and *partition-privacy*, that provide strong privacy guarantees with the introduction of very small noise. (Chapters 4,5).

4. The contribution of new, easily implementable, algorithms satisfying *contributor-privacy* that use ego-network style analysis to provide useful approximate results for queries that are too sensitive to perform under previous standards.(Chapter 4).

5. The contribution of easily-implementable algorithms applying *partition-privacy* to a variety of contexts; These techniques provide strong privacy guarantees for analyses which learn correlations from sets of graphs (for example, identifying patterns of behavior in student collaboration groups). (Chapter 5)

6. Application of these techniques to real world data, including online social networks. (Chapters 4,5).

7. A differentially private approach to determining statistical significance on paired-sample data, using the Wilcoxon Signed-rank Test (which is appropriate for non-normally distributed social network data). (Chapter 6).

8. The proposal of De Facto Privacy, a formal measure of the relative protection provided by deterministic approaches, which do not satisfy differential privacy. (Chapter 7)

In the next two chapters we will cover necessary background information and provide a survey of existing research in privatized social network analysis. In chapters 4-5 we will introduce and demonstrate new privacy-preserving algorithms for many social network applications. In chapter 6 we will propose a differentially private method for computing statistical significance on paired-sample data. And in chapter 7 we will define De Facto Privacy.

2. BACKGROUND

In this chapter, we provide a basic introduction to the two areas of research which are spanned by this dissertation. We explain the concept of differential privacy and describe how it has been applied to tabular data and network data, and we review core analysis techniques in social network analysis that will be referenced in the remainder of this work.

2.1 Introduction to Differential Privacy

Differential privacy, developed by Cynthia Dwork and her collaborators at Microsoft Research [22], states a mathematical guarantee of privacy that sufficiently well-privatized queries can satisfy; it is independent of any specific technique or algorithm. Consider a common sequence of events in social science research: a survey is distributed to individuals within a population; a subset of the population chooses to participate in the survey; individual information from the surveys is compiled into a data-set and some analysis is computed over it; the analysis may be privatized by the injection of random noise; and the final privatized result is released to the general public. Differentially-private queries offer a rigorous mathematical guarantee to survey participants that the released results will not reveal the nature of their participation in the survey.

We first introduce a few useful notations: I is set of individuals who contribute information to the data-set D_I (e.g., survey participants). The set of *all possible* data-sets is $\mathcal{D}$. We use $F : \mathcal{D} \to \Re^k$: to refer to the desired non-privatized analysis performed on a data-set and $Q : (\mathcal{D}) \to \Re^k$ to refer to the non-deterministic privatized implementation of F. Given a specific data-set D_I, evaluating the non-deterministic function $Q(D_I)$ produces a result $R \in \Re^k$ which has been privatized for general publication.

If R is the privatized query results that are released to the public, then R is the only evidence an attacker has about the nature of D_I. We introduce a possible-worlds model to understand how differential privacy works. We define D_I to be *true world* from which the analysis was taken. Two closely related variant definitions of differential privacy are in common use:

In the first variant definition (see Figure 2.1), we define any data-set that differs by the presence or absence of one individual to be a "neighboring" possible world: thus $D_{I-Alice}$ is the neighboring possible world of D_I in which *Alice* chose to *not* participate in the survey and D_{I+Fran} is the neighboring possible world of D_I in which $Fran$ chose *to* participate in the survey.

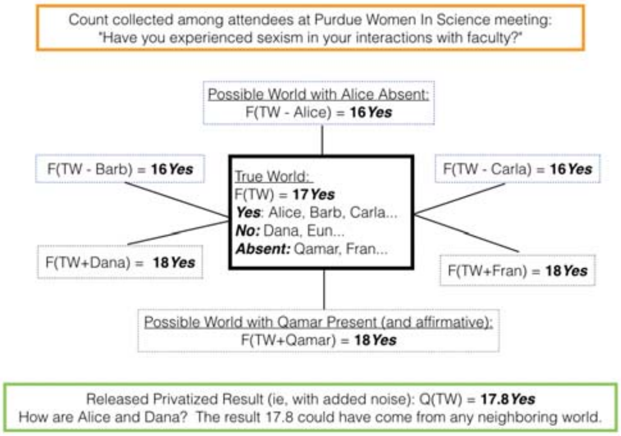

Count collected among attendees at Purdue Women In Science meeting:
"Have you experienced sexism in your interactions with faculty?"
Possible World with Alice Absent:
F(TW - Alice) = 16 Yes
F(TW - Barb) = 16 Yes
F(TW - Carla) = 16 Yes
True World:
F(TW) = 17 Yes
Yes: Alice, Barb, Carla...
No: Dana, Eun...
Absent: Qamar, Fran...
F(TW+Dana) = 18 Yes
F(TW+Fran) = 18 Yes
Possible World with Qamar Present (and affirmative):
F(TW+Qamar) = 18 Yes
Released Privatized Result (ie, with added noise): Q(TW) = 17.8 Yes
How are Alice and Dana? The result 17.8 could have come from any neighboring world.

Fig. 2.1.: Differential privacy adds noise to obfuscate individuals' effect on query results. In the first definition variant, an individual's presence in the data is hidden.

In the second variant (see Figure 2.2), we define a neighboring possible world to be any data-set that differs in the *value* of one individual: If in D_I *Bob* reports that he is Sad, then $D_{I(Bob \rightarrow Happy)}$ is the neighboring possible world of D_I in which *Bob* responded to the survey by stating that he was $Happy$. One significant difference between these variants relates to the number of individuals in the data-set: In the first variant the number changes between neighboring worlds while in the second variant it is fixed.

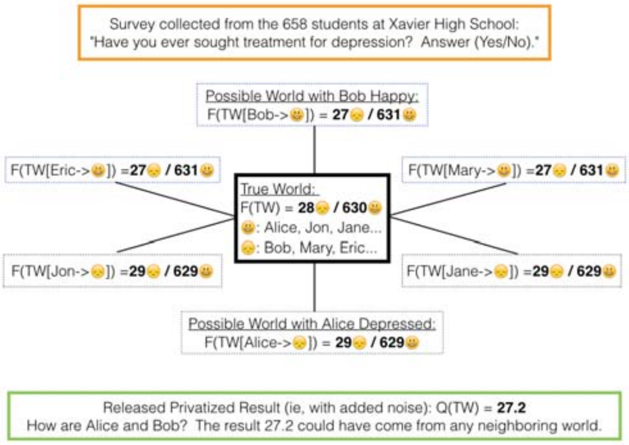

Survey collected from the 658 students at Xavier High School:
"Have you ever sought treatment for depression? Answer (Yes/No)."
Possible World with Bob Happy:
F(TW[Bob->😃]) = 27😞 / 631😃
F(TW[Eric->😃]) =27😞 / 631😃
F(TW[Mary->😃]) =27😞 / 631😃
True World:
F(TW) = 28😞 / 630😃
😃: Alice, Jon, Jane...
😞: Bob, Mary, Eric...
F(TW[Jon->😞]) =29😞 / 629😃
F(TW[Jane->😞]) =29😞 / 629😃
Possible World with Alice Depressed:
F(TW[Alice->😞]) = 29😞 / 629😃
Released Privatized Result (ie, with added noise): Q(TW) = 27.2
How are Alice and Bob? The result 27.2 could have come from any neighboring world.

Fig. 2.2.: Differential privacy adds noise to obfuscate individuals' effect on query results. In the second definition variant, an individual's value in the data is hidden.

In this work, we will most often make use of the first variant definition–it is the most commonly applied definition in existing privatized social network analysis research. This may be true in part because altering the 'value' of an individual in a network is not intuitively well-defined. If removing all of an individual's edges is an allowable alteration, the two variants are in fact equivalent for analyses such as triangle counts (section 2.3.2). Additionally, because it hides participation in the data-set entirely, the first variant is more applicable to sensitive social networks (such as sexual interaction networks) in which presence in the data-set itself is sensitive.

To satisfy differential privacy, we require that an attacker possessing the privatized results R be unable to determine with high certainty, between any two neighboring worlds, which world is the true one; R should be a plausible result from any neighboring world of D_I. With reference to the first variant definition, this intuitively implies the attacker is unable to use the privatized published survey results to guess with high probability whether or not *Alice* (or any other specific individual) took the survey, i.e., whether or not R is the result from an analysis of D_I or $D_{I-Alice}$. In the second variant, the attacker may be able to use the results to determine that Bob has taken survey, but he is prevented from learning whether Bob is happy or depressed (whether the true world is D_I or $D_{I(Bob \rightarrow Happy)}$).

Formally, we define Differential Privacy as follows:

Definition 2.1 ***Neighboring World:***

> Variant I: *Two data-sets D_{I1}, D_{I2} are* ***neighbors*** *if they differ by the addition or removal of exactly one individual:* $|I1 \cup I2 - I1 \cap I2| = 1$
>
> Variant II: *Two data-sets D_{I1}, D_{I2} are* ***neighbors*** *if they differ in data provided by exactly one individual: iff $I2 = (I1 - i) + j$ for arbitrary individuals i,j.*

Definition 2.2 ***Differential Privacy:*** *A randomized query*

$$Q : \mathcal{D} \rightarrow \Re^k$$

satisfies ϵ***-Differential Privacy*** *[18] if, for* any *two possible neighboring data-sets D_1, D_2 and* any *possible set of query results $R \in \Re^k$:*

$$\frac{Pr[Q(D_1) \subseteq R]}{Pr[Q(D_2) \subseteq R]} \leq e^{\epsilon}$$

Here ϵ is a small, positive value that controls the trade-off between privacy and accuracy, and is chosen by the person administering the privacy policy. To make the definition more intuitive, consider that if we set $\epsilon = ln(2)$, the above states that the result R is at most twice as likely to be produced by the true world as by any of its neighbors. Setting a smaller ϵ will provide greater privacy at the cost of additional noise, as we will demonstrate below.

The difference between the results from D_1 and any neighbor D_2 is the difference the privatization noise will need to obfuscate in order for the privatized results to not give evidence about whether D_1 or D_2 is likely to be the true world. The upper bound of this difference over $D_I \in \mathcal{D}$ is the *sensitivity* of query F.

Definition 2.3 ***Global Sensitivity:*** *The* ***global sensitivity*** *of a function* $F : \mathcal{D} \rightarrow R^k = A$ *is* [1]*:*

$$\Delta F = \max_{D_1, D_2} \|F(D_1) - F(D_2)\|_1$$

over all pairs of neighboring data-sets D_1*,* D_2*.*

Intuitively, the sensitivity of a query is the *greatest* possible separation between two neighboring worlds with respect to the query results. Under the first variant definition, this is the greatest possible impact that adding or removing an arbitrary individual from the data-set can have on the query results, over *any* possible data-set. Suppose our analysis F asks two questions: "How many people in I are failing?" and "How many people in I have fewer than 3 friends?" Then both answers can change by at most 1 when a single individual is added to or removed from I, and $\Delta F = 2$. If our analysis instead asks: "How many people in I are failing?" and "How many people in I are passing?" then at most *one* answer can change by at most 1, and $\Delta F = 1$. Note that, under this variant, histograms which partition the individuals of the data set into "bucket" counts have a sensitivity of 1: removing or adding an individual will change at most one bucket count by at most 1. This very low sensitivity makes histograms a useful tool in differentially private data-mining [22–24].

Note that under the second variant definition neighboring worlds are separated by changing one individual's value rather than their presence in the dataset, and this can result in different evaluations of function sensitivities. Consider Figure 2.2: Although the query here is a histogram, partitioning the students into mutually exclusive Happy and Depressed counts, altering one individual's value affects two counts by at most one, and this results in a slightly higher sensitivity of 2 in this case.

For the remainder of this work we will use the first variant definition; exceptions will be specifically noted.

We can create a differentially private query Q by adding noise to F that is calibrated to cover up ΔF [22]:

[1] The L_1-norm of $x \in \Re^n$ is defined as $\|x\|_1 = \Sigma_{i=1}^n |x_i|$.

Theorem 2.4 *If $F : \mathcal{D} \rightarrow \Re^k$ is a $k - ary$ function with sensitivity ΔF then the function $R = F(D) + Lap^k(\Delta F/\epsilon)$ is ϵ-differentially private, where $Lap^k(\lambda)$ is a k-tuple of values sampled from a Laplacian random variable with standard deviation $\sqrt{2}\lambda$.*

The standard deviation of the Laplacian noise values is $\sqrt{2}\Delta F/\epsilon$. Thus the noise will be large if the function is very sensitive, or if ϵ is small. If we set $\epsilon = ln(2)$ on a query with sensitivity $\Delta F = 2$, the standard deviation of our added noise will be close to 4.

It is important to note that ΔF is an upper bound taken across *all possible* pairs of neighboring data-sets; it is independent of the true world. Intuitively, this is necessary because noise values which are dependent on the nature of the true world may introduce a privacy leak themselves. For example, when querying the diameter of a social network, if Alice forms the only bridge between otherwise unconnected subgraphs in the true world, removing her node and edges from the data-set causes a difference of ∞ in the graph diameter. Noise values calibrated to this true world must be arbitrarily large (and, in fact, will obliterate the utility of the result). However, consider a neighboring *possible* world including Bob, who forms a second bridge between the subgraphs (see Figure 2.7); if this possible world were the true world, the difference in diameter caused by adding or removing a node would be finite, and if we calibrated the noise to that difference, it would be relatively small. If we chose our noise values based on the true world, an attacker could easily determine whether or not Bob was in the network: a result of $R = 300,453.23$ would imply Bob was absent, while the result $R = 4.23$ would indicate that Bob was present. To prevent this, global sensitivity is based on the worst-case scenario for the query, across all *possible* data-sets. In this example, this implies that diameter is a query too sensitive to be feasibly privatized using traditional differential privacy.

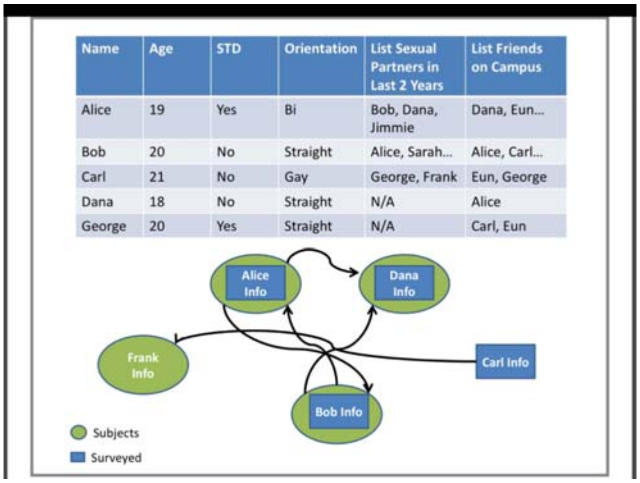
2.1.1 Smooth Sensitivity

Several sophisticated privatization techniques exist that calibrate noise to the true data-set, avoiding the worst-case upper-bound offered by global sensitivity. Consider an actual data-set D_{June12}; the *local sensitivity* of a function F on the data D_{June12} is the maximum change in F caused by removing or adding an individual from D_{June12}, analogous to computing the global sensitivity with D_1, D_2 restricted to D_{June12} and its neighboring possible worlds. In the example above, $diameter(G_{bob})$'s local sensitivity is small, while the local sensitivity of its neighbor $diameter(G_{alice})$ is very high: this jump in local sensitivities is what causes the threat to privacy described above. Since G_{alice} is created by removing one individual from G_{bob}, we will refer to G_{alice} as a *one-step neighbor* of G_{bob}, and consider a *k-step neighbor* of G_{bob} to be one created by adding or removing k individuals from G_{bob}. *Smooth sensitivity* is a technique which computes privatization noise based on both the local sensitivity of the true data-set, *and* the local sensitivity of all *k-step* neighbors scaled inversely by k, for all k [25]. The technique 'smooths' over the local-sensitivity jumps depicted in the alice-bob graph example. However, local-sensitivity based techniques satisfy a slightly weaker definition of differential privacy: (ϵ, δ)-*indistinguishability*. Privatization strategies which satisfy (ϵ, δ)-*indistinguishability* produce results $R \in \Re^k$ which satisfy a modified version of Definition 2.2 that includes an additive term: $\frac{Pr[Q(D_1) \subseteq R]}{Pr[Q(D_2) \subseteq R]} \leq e^{\epsilon} + \delta$, where δ is a negligible function of the data-set size n [2]. Additionally, in some cases computing the amount of noise required to privatize a given D_I may be infeasible. We will primarily focus on techniques which satisfy strict ϵ-differential privacy in this work, but we will reference existing smooth-sensitivity techniques where applicable, and we recommend consulting [26] for more information on this approach.

[2] A third variant, (ϵ, δ)-*differential privacy*, allows a constant value δ

2.2 Differential Privacy and Network Data

The definitions we introduced above for differential privacy, Definitions 2.1 and 2.2, implicitly assume all information about a data-set participant is provided by the participant themselves; protecting an individual's presence or submitted data value in the data-set then protects all the information regarding them. The situation changes when we ask survey participants to provide information about other individuals.



Name	Age	STD	Orientation	List Sexual Partners in Last 2 Years	List Friends on Campus
Alice	19	Yes	Bi	Bob, Dana, Jimmie	Dana, Eun...
Bob	20	No	Straight	Alice, Sarah...	Alice, Carl...
Carl	21	No	Gay	George, Frank	Eun, George
Dana	18	No	Straight	N/A	Alice
George	20	Yes	Straight	N/A	Carl, Eun

Fig. 2.3.: Unlike tabular data, participants in social network studies provide information about each other. This information may be incomplete or inconsistent.

We will refer to individuals who contribute their knowledge to the data-set as *participants*, and individuals who have information provided *about* themselves (by others) as *subjects*. Traditional differential privacy protects participants only, and in many cases subject privacy may be unnecessary. To clarify, we return to our view of a dataset as a survey of the real world: if a survey counts the students who attended the "Coffee with the Dean" event, the dean's privacy is probably not an issue. By contrast, a study that counts students who report having sexual relations with the football captain exposes extremely sensitive information about its subject. Social networks are often collected from populations of interest by having participants list the full names of their friends within the population; these relationships form directed network edges leading from the participant's node to the nodes of each of her friends [27]. In this case, a participant's real world friends are subjects of the participant's "survey data", but the participant herself may also be the subject of some of her friends' survey data (if they also participate in the social network). This presents a complex situation for applying differential privacy. Figure 2.3 illustrates an example.

The core of the differential privacy guarantee is that the privatized result R is difficult to attribute to the true world vs. one of its neighboring possible worlds. Adapting differential privacy to networked data amounts to deciding what we mean by "neighboring worlds" in this context. There are several possibilities; each one provides a different level of privacy guarantee and deals with a different type of "gap" between worlds. As always, there is a trade-off between privacy and utility: in general, the stronger the privacy guarantee, the more noise will be required to achieve it and the less useful the privatized results will be. We will describe two network privacy standards, *node-privacy* and *edge-privacy*, that have appeared in the literature.

In subsequent chapters, we will propose two novel standards, *contributor-privacy* and *partition-privacy*, that require less noise than existing standards; give a reasonably strong guarantee of privacy similar to traditional differential privacy; and enable certain queries that under existing standards required levels of noise that rendered results meaningless.

2.2.1 Node-Privacy

The Alice-Bob graph example referenced in Section 2.1.1 and Figure 2.7 implicitly assumes this privacy standard: In node privacy, if the true world is a given social network G, the neighboring possible worlds are ones in which an arbitrary node, and *all* edges connected to it, are removed from or added to G. Formally,

Definition 2.5 ***Node-Privacy:*** *A privatized query Q satisfies* ***node-privacy*** *if it satisfies differential privacy for all pairs of graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ where $V_2 = V_1 - x$ and $E_2 = E_1 - \{(v_1, v_2)|v_1 = x \vee v_2 = x\}$ for some $x \in V_1$*

Note the two equivalent interpretations of this definition: G_1 is equal to G_2 after the addition of an arbitrary x, or G_2 is equal to G_1 after the removal of an arbitrary x.

This privacy guarantee completely protects *all* individuals, both participants and subjects. An attacker in possession of R will not be able to determine whether a person x appears in the population at all. Although this is an natural adaptation of differential privacy to social networks, it also places *extremely* severe restrictions on the queries we are able to compute, as we will demonstrate in Chapter 3, in many cases, node-privacy may be an unnecessarily strong guarantee. Figure 2.4 depicts neighboring worlds in a triangle-count (see Section 2.3.2) under node-privacy.

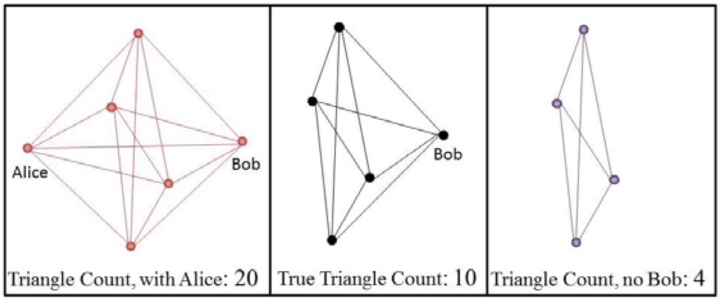


Fig. 2.4.: Node-sensitivity of triangle-counts is a function of n, and thus is unbounded in general.

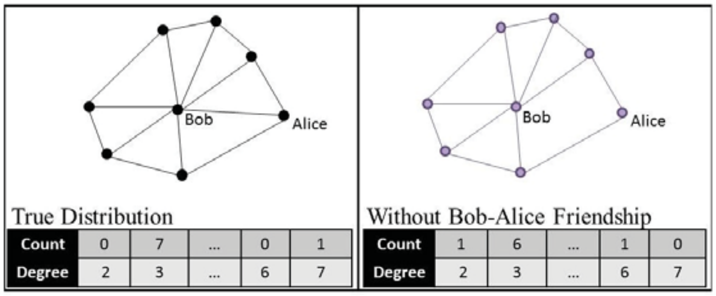
2.2.2 Edge-Privacy

In edge-privacy, if the true world is the social network G, neighboring possible worlds are ones in which k arbitrary edges are added or removed from G. Formally,

Definition 2.6 ***Edge-Privacy:***

A privatized query Q satisfies ***edge-privacy*** *if it satisfies differential privacy for all pairs of graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ where $V_1 = V_2$ and $E_2 = E_1 - E_x$ where $|E_x| = k$*

An attacker in possession of R won't be able to determine with high certainty whether individuals x and y are friends, and an individual node in the graph can plausibly deny the existence of up to k of its friendships with other nodes. Single edge privacy, with $k = 1$, is the standard most often used in existing literature on differentially private graph analysis. This is a weaker guarantee than node-privacy: high-degree nodes may still have an easily identifiable effect on query results, even though their individual relationships are protected. However, this is a sufficiently strong privacy guarantee for many contexts, and enables many more types of queries to be privatized than the severely-restrictive node-privacy. Figure 2.5 depicts neighboring worlds in a degree distribution (see Section 2.3.2) under edge-privacy.



True Distribution

Count	0	7	...	0	1
Degree	2	3	...	6	7

Without Bob-Alice Friendship

Count	1	6	...	1	0
Degree	2	3	...	6	7

Fig. 2.5.: Edge sensitivity of degree distribution queries is 4: at most four values can change by one when a node is added or removed.

2.3 Social Network Analysis Background

2.3.1 Introduction

In this section we will review the social network analysis techniques which will be discussed in the remainder of this work. While we focus primarily on those network analysis techniques that have so far proven at least partially amenable to privatization, we will also briefly sketch the properties of network data that deter the privatization of other types of analyses. Intuitively, privatizable network characteristics are those that tend to remain approximately consistent under small changes in the network data; these are the same characteristics that are robust to small errors in data-collection.

Social networks offer a useful abstract model of individuals (represented as graph nodes) and the relationships that connect them (represented as edges between nodes). Social network analysis can be a very important tool, allowing researchers to learn about populations by identifying meaningful patterns in their social networks. For example, understanding how well-connected a network is can aid in the development of word-of-mouth marketing campaign: How quickly will word of a product spread? Similar analysis is useful in epidemiology, predicting spread of a disease, or in learning analytics, studying how students interact and collaborate. One of the earliest social networks used in CS research is given in Figure 2.6, originally published in 1977 [7].

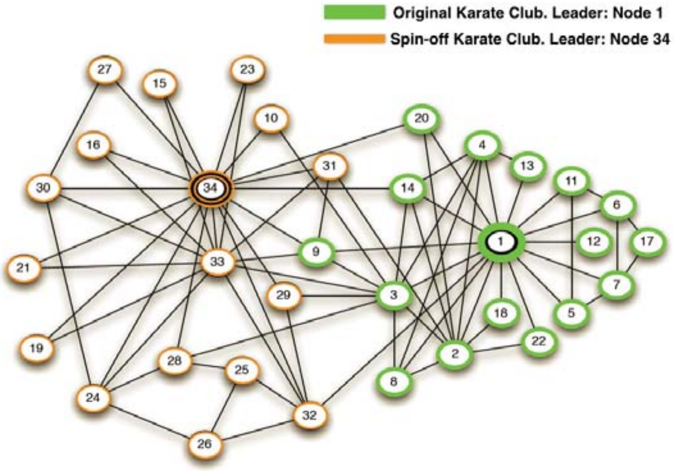

🟩 Original Karate Club. Leader: Node 1
🟧 Spin-off Karate Club. Leader: Node 34
27
15
23
10
20
16
4
13
31
30
34
14
11
6
1
12
17
9
21
33
3
29
7
5
18
22
19
28
25
2
8
24
32
26

Fig. 2.6.: An anonyzmized social network collected over the 34 members of a university karate club, shortly before a schism caused the group to split in two.

2.3.2 Social Network Analysis Review

We now provide a high level introduction to common terminology and techniques in Social Network Analysis. We divide the field into six broad categories, with respect to objective and types of computation: Edges and Degrees, Triangle and Subgraph Counts, Centrality and Path-length Measures, Community Detection, and Graph Models. For each topic we will define basic terminology, outline common analysis techniques, and discuss any challenges to privatization. We will reference these categories throughout the remainder of this work.

It is important to note that social network analysis studies human behavior through abstract data-structures which encode real word social structures. Objectives include learning about the relative social importance of individuals in the network, how information propagates through a population, and how individuals form into communities. For each objective, there are a variety of possible analysis techniques; due to the ambiguity inherent in human social behavior, there is often no single definitive correct result or best analysis technique. We review a selection of the most commonly-used techniques below.

Edges and Degrees

The **degree** of a node is the number of edges connecting to it, in a social network this often represents the number of friends the individual has in the network. If a graph is **directed**, edges may be uni-directional arrows pointing from one node to another (for example, if Amy lists Bob as a friend, but Bob does not return the favor, the edge will appear in the graph as an arrow leading from Amy to Bob). An edge leading out from a node is referred to as an **outlink**, and an edge leading into a node is an **inlink**; a mutual edge is **undirected** and can be counted as both an outlink and an inlink. The count of a node's outlinks is its **out-degree** (**in-degree** is defined analogously). In directed graphs, **reciprocity** metrics measure the extent to which the edges in the graph tend to be mutual [28]. We see that the karate club graph is undirected, and that nodes 1 and 34 have high degree. The **edge density** of a graph is the total number of edges divided by the total number of nodes; this is also referred to as the **average degree**.

In some graphs, edges are labeled with numbers that indicate the relative 'strength' of the edge connection (for example: the number of emails exchanged between the node individuals); these labels are referred to as the **weight** of the edge. Edges may also be labeled with relationship properties: In a **signed** graph, positively labeled edges may be used to represent friendships while negatively labeled edges represent enemies. Nodes can be labeled with properties of the individual they represent, such as gender, career status, or course grade. In graphs with node labels, **homophily** metrics measure the extent to which nodes that share label values tend to be connected, relative to nodes that do not share label values [29].

A node i's **neighborhood** is the subgraph whose nodes consist of i and i's friends, and whose edges consist of all edges that connect these nodes. For example, in the karate club graph, the neighborhood of node **17** has nodes: **17,6,7**, and edges: **(17,7),(17,6),(7,6)**. This is also sometimes referred to as a node's $ego - network$.

The degree distribution of a graph is a histogram partitioning the nodes in the graph by their degree; it is often used to describe the underlying structure of social networks for purposes of developing graph models and making similarity comparisons between graphs [30]. Intuitively, a graph with a few very high degree nodes and very many low degree nodes will describe a more hierarchical social organization than a graph with a more egalitarian degree distribution (although almost all large social networks possess a power-law degree distribution).

Triangle Counts and Subgraph Counts

Triangles, instances in which two of an individual's friends are themselves mutual friends, indicate social cohesion in the network. **Global triangle counts**, a count of all the triangles in the network, are often used to study and compare networks and as parameters in social network models.

Additionally, we can look at patterns with triangles on the local level. An individual's **local clustering co-efficient** is the ratio between the number of triangles the node participates in and the maximum possible number of triangles for a node of that degree [31]. A **clique** is a graph, or sub-graph, in which all possible edges exist; in a friendship graph, this implies all nodes in the set are friends with all other nodes in the set. The local clustering coefficient for individuals in a clique is 1, provided there are no edges connecting to individuals outside the clique. Distributions of local clustering coefficients across the network (analogous to degree distributions) can give more detailed information for comparing networks [32].

Finally, counts of other subgraphs such as stars, or squares, are also used as graph statistics for graph similarity comparisons between networks [33].

Centrality and Path-length measures

A **path** is a sequence of edges connecting two nodes; for example a path between node **27** and node **24** in the karate graph in figure 2.6 is the one comprised of the two edges **(27, 30)** and **(30, 24)**. The **length** of a path is the number of edges comprising that path (e.g., the length of the given path between nodes **27** and **24** is 2). The **shortest path** between two nodes is the path between them with minimal length (e.g., the given path between nodes **27** and **24** is the shortest path between them). The **distance** between two nodes is the length of the shortest path connecting them, so the distance between node **27** and **24** is 2.

Centrality measures attempt to gauge the relative "importance" of specific individuals within the social network; they may be studied on a per-node basis, identifying influential members of the community, or as distribution scores providing information about the overall behavior of the social network [34]. In the karate club network (Figure 2.6), the individuals represented by nodes **1** and **34** had high importance in

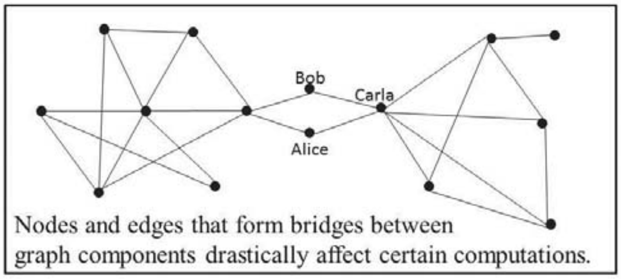

Bob
Carla
Alice
Nodes and edges that form bridges between graph components drastically affect certain computations.

Fig. 2.7.: Removing one node or edge from a graph can change path lengths catastrophically.

the social group the graph abstracts; They were the leaders of two karate clubs that formed after the schism. The simplest centrality measure is node degree: nodes with high degree are more likely to be influential in the network, (note that nodes **1** and **34** have high degree). However, other centrality measures take into account more detailed information from across the network: **betweenness** scores individuals by the number of shortest-paths between other pairs of nodes across the network that pass through them, and **closeness** scores nodes by the sum of their distances to all other nodes in the graph.

The two more complex centrality measures, **betweenness** and **centrality**, present difficulties for traditional approaches to privacy in social networks. Clearly, it is impossible to release a named list of influential individuals under anonymity or differential node-privacy. But even distributions of centrality scores can be very sensitive, under both differential node- and edge-privacy, due to the role of **bridges** in the graph. Removing a node, or edge, that forms the only connection between two otherwise disconnected subgraphs will have a catastrophic affect on path distances in the network, causing finite distances to become infinite, and this will drastically alter betweenness and closeness scores (see Figure 2.7). Privatization methods that delete or add edges to privatize network structure, such as noisy anonymity and k-anonymity (introduced in the next chapter), may inadvertently add or delete bridges. In general, privatizing traditional centrality measures, or any metric that relies significantly on path lengths, remains a very difficult problem for privacy. However, special cases exist; we propose practical solutions for several such cases in Chapter 5.

Community Detection

People form social groups naturally, and this is reflected in social networks as clumps of nodes which are more densely connected to each other than to the outside network. We refer to these as social **communities**. For instance, the karate club network has two distinct social communities; these communities later actually separated into two clubs. There are a variety of methods for identifying communities in a network [35]. Two simpler approaches are: **Modularity**-based techniques, which involve identifying a subset of nodes that are more strongly attached to each other than to the outside graph, and the **Girvan-Newman** algorithm, which splits graphs into communities by removing bridge nodes (nodes with high betweenness) that connect them. More complex techniques consider the eigenvalues of the connection matrix for a network. Community detection is a difficult problem for privatization, similar to the difficulties found in centrality and path-length analyses (recall that the betweenness of a node may vary catastrophically with the addition or removal of another node or edge). However, recent work has made progress here under edge-privacy; we present an overview of these approaches in Chapter 3.

Graph models

Graph models attempt to abstract the underlying social interaction patterns that produce networks in populations [36]. In general, a graph model takes in parameters that describe a real network (such as the network's degree distribution, average degree, or triangle count), and then can be used produce a randomized synthetic graph that shares these properties. Models enable formal analysis of network properties by providing a simple, well-defined abstraction which allows for drafting and proving hypotheses about network properties. They are also useful for producing privacy-preserving synthetic data-sets which share similar properties to the real, sensitive data-sets. In theory, these privatized synthetic networks may be safely shared and studied in place of the real data; in practice, they may lose important properties from the original graph, such as clustering patterns. Many different graph models exist, each with advantages and disadvantages. A variety of differentially private models have been proposed by privacy researchers; we'll discuss these in Chapter 3.

3. RELATED WORK

As the field of social network analysis has advanced in the decades since the publication of the Karate Club graph, the problem of privacy-preserving social network analysis has received increased attention. A diverse variety of techniques have been proposed. We summarize these techniques in chronological order, corresponding to the development of increasingly rigorous privacy guarantees.

We believe that several properties are important for a privatization technique to be practically usable in real world contexts:

- **Guaranteed Privacy**: It must provide a well-defined privacy guarantee to individuals in the data-set.
- **Maintain Utility**: It must enable privatized analyses to produce results with a reasonable level of accuracy.
- **Practically Adoptable**: To encourage adoption it must not impose a significant burden in computing power or mathematical expertise in comparison to the non-privatized analysis it replaces.

Many of the existing proposed techniques satisfy one or two of these requirements, but fail to satisfy all three.

3.1 Simple Anonymity and De-anonymization

Initial attempts to protect the privacy of individuals in social network studies used simple anonymization, as in the Karate Club graph (figure 2.6). This remains a common standard in practice in social science research today [1–6]. However, simple anonymity is subject to de-anonymization attacks when an attacker has access to outside information about a population. For instance, knowing how many friends an individual has may be sufficient to identify their node in a small anonymized graph. Once a few individuals have been re-identified, some friends of these individuals may become identifiable, and the de-anonymization proceeds by the same means out across the graph. Public online social networks (OSN) such as Twitter, IMDB, and Pinterest help attackers get access to the outside information necessary for these attacks. When the correct names have been mapped to the nodes in an anonymized graph, any privacy protection the publisher hoped to provide is invalidated. Attackers may discover sensitive information in node-labels (such as weight, grade, or disease status), in edge-labels, and in the existence of previously unknown relationship edges between individuals (problematic in sensitive data-sets such as sexual relationship networks). In larger graphs, identifying small unique subgraphs in the anonymized graph is sufficient to begin the de-anonymization procedure [16].

A simple step to address this vulnerability is to add 'noise' to the graph structure: randomly add and delete edges and nodes in the anonymized graph to obfuscate the true graph structure [37]. However, the effect of this noise is often insufficient to hide unique structures in the graph. De-anonymization techniques are sufficiently powerful to attack very large real world graphs, even in the presence of structural noise and some sophisticated structural anonymization techniques [16, 17, 38]. Massive de-anonymization attacks have been executed on the Netflix data-set and an anonymized Twitter graph, uncovering the actual individuals associated with the anonymous data (with significant negative consequences in the Netflix case). Simulated de-anonymization attacks have also been demonstrated on the Enron Email and Facebook data-sets, using synthetic user IDs [8, 16, 17, 38].

3.2 K-anonymity and Related Approaches

De-anonymization attacks demonstrate that individuals with unique social structures are vulnerable in simply anonymized networks, and may remain vulnerable even after random noise is added to the graph structure. *K-anonymity* is an approach to privatization, developed by Latanya Sweeney, that restricts the presence of unique individuals in published data-sets [39]:

Definition 3.1 ***K-Anonymity:*** *A set of records V with attributes A satisfies* ***k-anonymity*** *if for every tuple $v \in V$ there exist at least $k1$ other tuples $v_{i1}, v_{i2}, ..., v_{i(k\text{-}1)} \in V$ such that $\forall A_q \in A$:*

$$v_{i1}.A_q = v_{i2}.A_q = ... = v_{i(k\text{-}1)}.A_q$$

where $A_q \in A$ is a quasi-identifying attribute.

K-anonymity relies on the data-publisher to decide which attributes of the data are sensitive (quasi-identifying) and might be used to re-identify individuals. In tabular data, personal information attributes such as zip code or date of birth are often considered to be quasi-identifying. There are a variety of interpretations for what comprises a sensitive attribute in a social network. In general, greater levels of anonymity protection require more complex algorithms to achieve and have a greater impact on the utility of the data.

Researchers have looked at *k-degree* anonymity, ensuring that no node had a unique degree that could be used to identify it [40]. However, if an attacker had additional information about an individual's friends, she might still be able to identify him by his unique neighborhood graph. Researchers proposed *k-neighborhood* anonymity to address this, ensuring that no node had a unique neighborhood [1] graph [41]. Neither of these approaches would guarantee protection if the attacker had possession of a larger subgraph, such as two friends with a unique *pair* of degrees, or a set of several connected neighborhoods which formed a unique subgraph. Thus, even stronger forms of k-anonymity have been proposed. *K-automorphism* and *k-confusion* anonymity both ensure that each node is indistinguishable from at least k others (in terms of edges), for an arbitrarily sized subgraph: for example, consider a set of k

[1] Recall from Chapter 2 that an individual's *neighborhood* is comprised by the individual's node, its friends ('neighbors'), and all edges that connect these nodes.

leaf nodes connected to the same central node, or a clique of size k. These approaches require extremely complex algorithms to achieve and the resulting privatized graphs do not necessarily share many features with the original data [20, 42].

Additionally, the *k-anonymity* approach to privatization has a few well-known faults which also apply in the social network analysis context. Consider a graph tracing the spread of an STD. If an attacker knows that Carl has degree 18, and every node of degree 18 is labeled as having been infected with syphilis, the fact that there are at least k nodes of degree 18 in the graph does not provide Carl with meaningful privacy protection.

One approach that has been proposed to address this weakness is *l-diversity* [43]. L-diversity requires that each sensitive attribute category also has a diverse set of data values: at least l different values must be "well-represented" for each equivalence class of quasi-identifying attributes. However, l-diversity further decreases the utility of the analysis by altering the distribution of the data. Additionally, the privatization algorithm may leave a recognizable mark on the data, which can allow the privatization steps to be undone. Consider an attacker who is aware that a graph has been released with privacy parameters $k = 12$, $l = 2$, and *representation-threshold*$= 3$, requiring that each quasi-identifying category contain at least 12 individuals and both the values "syphalitic" and "healthy" must be represented in the category at least three times. If the attacker then observes that in the category of degree-18 nodes there are precisely 9 syphilitic and 3 healthy nodes, she may be able to infer the original true data values. [15].

3.3 Differential Privacy

Differential privacy offers a formal guarantee of individual participant's privacy that is not conditional on the attacker's background knowledge. As we described in chapter 2, there are two existing standards of differential privacy on social networks: edge privacy and node privacy. In edge privacy, neighboring worlds vary by one relationship; in node-privacy they vary by one node and all of its edges. In general, edge-privacy provides a relatively weak privacy guarantee, while the much stronger node privacy is hard to attain without significantly affecting utility.

Recall that, in contrast to anonymity methods, which attempt to privatize raw data-sets, differential privacy generally focuses on either privatizing the output of functions over data or producing entirely synthetic data-sets. In the following sections we will demonstrate how node-privacy and edge-privacy apply to our categories of social network applications (as described in Chapter 2), and discuss what has been accomplished so far in each category. In the subsequent two chapters we will introduce our own adaptations of differential privacy to social network data, which we refer to as contributor-privacy and partition-privacy, and we will discuss how these adaptations resolve several of the difficulties with node-privacy and edge-privacy that are described below.

3.3.1 Triangle Counting

Node-Privacy

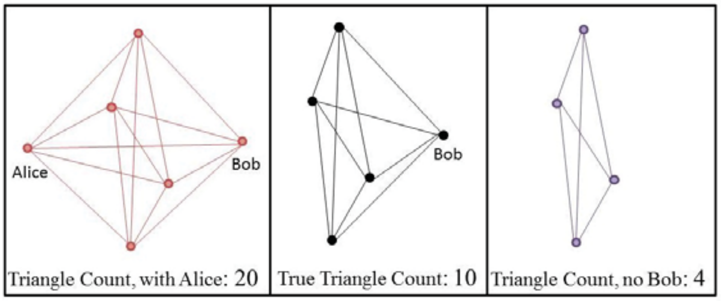


Fig. 3.1.: Node-sensitivity of triangle-counts is a function of n, and thus is unbounded in general.

Differentially private triangle counts are not feasible under simple node-privacy. In the worst case, adding a node to a complete graph of size n (a graph containing all possible edges), will introduce $\binom{n}{2}$ new triangles (Figure 3.1). Since the change is dependent on the size of the graph, the global sensitivity of the query in general is unbounded: it is impossible to compute a finite global upper-bound (see Section 2.2).

To address this issue, another approach has been proposed [19], using ideas similar to the smooth sensitivity approach described in section 2.1.1. If it is publicly known that the maximum degree of a graph is d, then removing or adding a node can affect the triangle count by at most $\binom{d}{2}$. Furthermore, any graph whose maximum degree is greater than d will have a *k-step* neighbor, for some k, whose maximum degree will be d (i.e., high-degree nodes can be removed until the maximum degree of the graph falls within the threshold). On generally sparse graphs with few nodes above degree d, the number of triangles in this bounded-degree neighbor graph will be a close approximation of the correct answer. The operation of finding the low-degree neighbor incurs its own sensitivity cost, but privacy can be still achieved at a sensitivity cost in the range $O(d^2)$ [19]. While this is untenable for large social networks, networks with low maximum degrees may successfully apply node-privacy to their triangle counts using this method.

Edge-Privacy

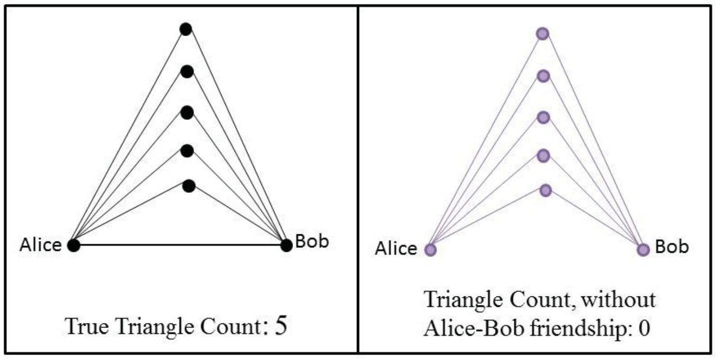


Fig. 3.2.: Edge-sensitivity of triangle-counts is a function of n, and thus is unbounded in general.

For similar reasons to node privacy, edge privacy is also not feasible for triangle-counts. In the worst case, removing an edge from a graph with n nodes can remove $n-2$ triangles (Figure 3.2). Since the sensitivity is a function of the graph size, it is unbounded in general.

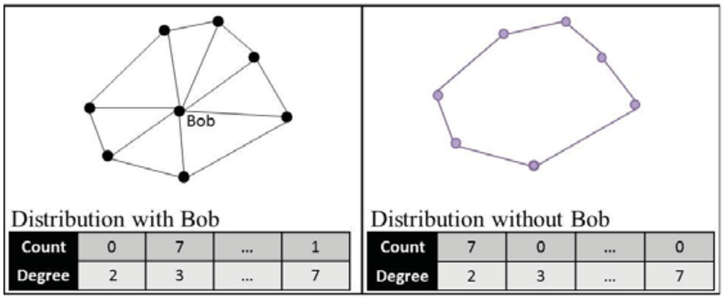
However, the local sensitivity of this query under edge-privacy, the sensitivity over a specific data-set, is bounded (even when the maximum degree is unbounded). Consider two nodes, a and b, that have k wedges (paths of length 2) connecting them, as in figure 3.2. If G is a graph in which no pair of nodes has more than k wedges connecting them, then adding an edge to G will create at most k triangles, and removing an edge will delete at most k triangles. We can apply smooth sensitivity techniques to take advantage of this in cases where k is not large, and thus attain the somewhat weaker (ϵ, δ)-*indistinguishable edge-privacy* (see Section 2.1.1). However, real-world social networks are transitive (if two people share a mutual friend, they're much more likely to be friends with each other) which can cause large values of k in practical applications of this technique. When k is large, even instance-based noise addition may introduce error of a factor of 10 or greater in analysis results [25]. The global clustering coefficient, which has the practical effect of normalizing the global triangle count using information taken from the degree distribution, can also be privatized with a careful application of smooth-sensitivity edge-private techniques. In this context, the effect of the noise is reduced [44].

3.3.2 Additional Approaches

In Chapter 5 we introduce partition-privacy, which considers social network analysis applications that operate on a set of graphs. Shen et. al. consider a related context in mining frequent small graph patterns over graph databases, such as those used in bio-informatics applications. They adapt a tabular-based frequent item-set approach to the problem, which incidentally satisfies partition-privacy [45].

3.3.3 Degree Distribution

Node-Privacy



Distribution with Bob

Count	0	7	...	1
Degree	2	3	...	7

Distribution without Bob

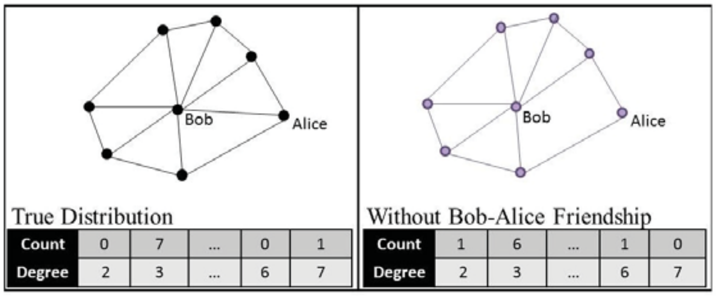
Count	7	0	...	0
Degree	2	3	...	7

Fig. 3.3.: Node sensitivity of degree distribution queries is a function of n, and thus is unbounded in general.

Although degree distributions are represented as histograms, the sensitivity is not small under node-privacy because one node affects multiple counts in the distribution: removing a node from the graph reduces the degree of all nodes connected to it. A node with k edges can affect a total of $2k+1$ values of the distribution (Figure 3.3). In the worst case, adding a node of maximal degree will change $2n+1$ values, and since this sensitivity is dependent on n, it will be unbounded in general (see Section 2.2).

Edge-Privacy

Edge-privacy is feasible for degree distributions. Removing one edge from the graph changes the degree of two nodes, and affects at most four counts (Figure 3.4). Under k-edge-privacy, the sensitivity is $4k$. With a sufficiently large graph, this is a negligible amount of noise, and the utility of this technique has been successfully demonstrated [24]. Recent work has improved results by considering the set of feasible degree sequences (ones which can be produced by graphs, without self-loops or multi-edges) and post-processing privatized results to fall within this set [46].



True Distribution

Count	0	7	...	0	1
Degree	2	3	...	6	7

Without Bob-Alice Friendship

Count	1	6	...	1	0
Degree	2	3	...	6	7

Fig. 3.4.: Edge sensitivity of degree distribution queries is 4: at most four values can change by one when a node is added or removed.

3.3.4 Centrality and Path-length Measures

Social network analyses such as centrality, clustering and path-length measures, which can be drastically affected by the existence of bridges in the data, are very challenging to privatize (as described in the previous chapter). Beyond the novel contributor and partition-privacy techniques we introduce in subsequent chapters, little existing work addresses the privatization needs of these analyses.

3.3.5 Community Detection

By contrast, progress has been made with attaining edge-privacy in the eigenvalues of the matrix representation for a graph. Spectral clustering methods use a graph's eigenvalues to identify dense clusters of nodes and edges in the graph. Recent work has allowed researchers to privately publish edge-private approximations of graphs that preserve clustering structure [21, 47].

3.3.6 Graph Modeling and Social Recommendations

Several researchers have proposed differentially private approaches to creating graph models: randomized synthetic graphs that are generated to be similar to a true, private, social network and thus can be studied safely in place of the original graph.

The Stochastic Kronecker graph model has been privatized under edge-privacy [48], as have exponential random graphs [49]. Several other groups have developed their own models that satisfy differential edge-privacy [50–53].

3.4 Summary

Returning to our initial three desired properties for privacy-preserving network analysis techniques, we note that although considerable research has been done in this topic in recent years, there still remains room for improvement:

Guaranteed Privacy: Anonymity approaches (including simple anonymity and k-anonymity) do not provide any broad guarantee of individual privacy which is not conditional on attacker's background information. Differential edge-privacy provides a broad guarantee for a limited amount of individual information (protecting any one relationship edge), while differential node-privacy provides a very strong guarantee of individual privacy (protecting all information regarding an individual's attribute values or relationships in the network).

Maintain Utility: Simple anonymity, some approaches to k-degree anonymity, and some applications of differential edge-private analysis produce high-utility results (in which added noise is non-existent, or has little practical impact on privatized results). By contrast, more advanced k-anonymity techniques (such as k-confusion), differential node-privacy, and high-sensitivity edge-private analyses may produce results with low utility (in which added noise or altered data causes privatized results to differ significantly from non-privatized data).

Practically Adoptable: Simple anonymity and some approaches to k-degree anonymity are both easily explained and easily implemented by data analysts with limited expertise in privacy. By contrast, more advanced k-anonymity techniques (such as k-confusion) and many differential privacy techniques require complex implementations and may not be efficiently computable.

In summary, there is a need for easily implemented analysis techniques which both provide high utility and satisfy robust privacy guarantees. In subsequent chapters we will introduce a variety of techniques which address this need.

4. CONTRIBUTOR PRIVACY

We now introduce a new application of differential privacy to network data, *contributor-privacy*, which provides a privacy guarantee very similar to the guarantee given to individuals in tabular data. Contributor-private algorithms use ego-network style analysis (focusing on nodes and their immediate neighbors [54]) to provide useful approximate results for queries that are too sensitive to perform under the previous differential privacy standards, node-privacy and edge-privacy.

Recall from Chapter 2 that we refer to individuals whose knowledge is contributed to the data-set as survey *participants*, and individuals who have information provided *about* themselves (by participants who have knowledge of them) as *subjects*. Differential privacy on tabular data guarantees survey participants that the privatized results will give very little evidence about whether they participated or not. Contributor-privacy offers the same guarantee to participants in network data.

4.1 Definition

Define PoI to be the Population of Interest, and $C \subseteq PoI$ to be the set of people who contribute information to the data-set (the survey participants). Recall that an ego-network is the vertex-induced subgraph of a node's neighborhood; it consists of the node, the node's direct friendships/friends, as well as the relationships among those friends.

For $i \in C$, define $d_i = (Info(Ego_i), Info(i))$ to be the information contributed by individual i to the data-set. This will include information about themselves, $Info(i)$, and about others within their ego-network $Info(Ego_i)$. Note that while $Ego_i \subseteq PoI$, it is not necessarily true that $Ego_i \subseteq C$. We use $D = \{d_i | i \in C\} = \{Info(Ego_i), Info(i) | i \in C\}$ to refer to the set of d_i comprising the data-set.

Definition 4.1 ***Contributor-Privacy:*** *A privatized query*

$$Q : \mathcal{D} \rightarrow \Re^k$$

satisfies ***Contributor-Privacy*** *if, for all $R \subseteq range(Q)$, and all pairs of data-sets $D_1 = \{Info(Ego_i), Info(i) | i \in C_1\}$, and $D_2 = \{Info(Ego_i), Info(i) | i \in C_2\}$ where $C_1 = C_2/i$, for some $i \in C_1$:*

$$\frac{Pr[Q(D_1) \in R]}{Pr[Q(D_2) \in R]} \leq e^{\epsilon}$$

This privacy guarantee protects the data contributed by data-set *participants*, using a standard conceptually similar to the definition of differential privacy over tabular data. If the true world is a social network G and the survey asks each individual i to list their gender (their node label, $Info(i)$) and who they believe is their friend (their outlinks, $Info(Ego_i)$), then the neighboring possible worlds are ones in which an arbitrary node and all of the information it contributed to the network (its node label and its outlinks) are removed from or added to G. An attacker in possession of the privatized results R won't be able to determine whether a person i supplied their data (submitted a survey) to help produce the graph. This privacy guarantee is strictly weaker than node privacy, but compares well with single edge privacy for many queries. Any participant can plausibly deny its out-links, or, equivalently, any participant can plausibly deny one in-link from another participant node. Analogous to k-edge-privacy, we can also provide k-contributor-privacy by considering neighboring worlds that differ from the true world by the information-contribution of up to k members of the graph. With the d_i above, 2-contributor-privacy allows two nodes to *simultaneously* deny all out-links, and as a result, this enables a complete mutual edge to be protected (providing single-edge privacy in addition to out-link privacy). In general, a k-level privacy guarantee can be satisfied by scaling the added noise by k.

However, contributor-privacy can also be generalized to cases in which d_i includes information beyond a simple list of i's perceived friends; any information that i can contribute to help us learn about PoI can be collected and privatized under contributor-privacy. We provide several practical examples below.

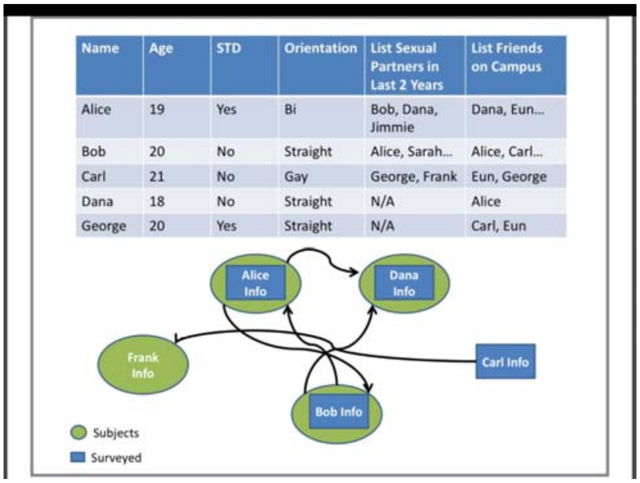
4.1.1 Privacy Analysis

Contributor-privacy guarantees that a participant submitting their survey to the data-set will have this contribution obfuscated in the results of any privatized query. However, in the context of network data, individuals customarily provide information about other individuals as well as themselves. If an edge or a node in a graph is well-known, what level of protection is offered by contributor-privacy? As described in Chapter 2: traditional differential privacy does not protect subjects, node-privacy does protect subjects, and edge-privacy offers very limited protection to subjects. In this section we will briefly discuss the protection offered to subjects by contributor-privacy, first examining the extremes of contributor-privacy protection and then briefly outlining an important factor in the general case. For a detailed understanding of subject privacy, we note that the obfuscation provided to subject data in noisy contributor-private algorithms is strictly greater than the obfuscation provided by deterministic algorithms that publish true data values. In Chapter 7, we will propose and demonstrate a metric for measuring the relative privacy provided by deterministic data analysis algorithms.

In the worst case, contributor-privacy may provide very little protection to an extremely well-known piece of network data. For example, a population of high school students is likely to be well-informed about a relationship edge between the school's head cheerleader and the football team captain. Surveying these students about the existence of this edge, and then adding to the result laplacian noise with parameter $(1/\epsilon)$, will produce a publishable result which satisfies contributor-privacy. This result protects the survey-takers, allowing any student to deny having reported on the relationship. However, as the magnitude of noise added will likely be much smaller than the number of students surveyed, the result will provide very little privacy protection for the two star athletes who serve as the survey's subjects. In this worst case, we might argue that this relationship information was already in the public sphere before the survey was conducted; privacy protection for the subjects would be of little value. However, publicly publishing this data will allow the information to be shared beyond the scope of the original population, where it may be less well-known. This presents an ethical issue that must be considered in any data-mining context: privatized data-mining allows aggregate facts about populations to be published while protecting individual data, but if the aggregate facts themselves are considered sensitive (for example: rates of drug-use in a particular university dorm, or rates of AIDS infection within a small neighborhood), then the data-mining results should not be published.

In the best case, contributor-privacy will provide strong protection for subject data that is not well known. In our previous example of a high school student population, we might posit a secret tryst between the indicated persons, which they alone report. In this case, contributor-privacy is effective at subject protection. In fact, analyses which satisfy 2-contributor-privacy will completely protect this edge in the data (standard 1-contributor private analyses will provide subject protection with parameter $\epsilon_{subject} = 2\epsilon_{participant}$).

The above covers two extreme cases, in which a subject data element is either completely private or publicly known. To understand the general case, we consider the effect of an attacker's level of certainty about the true data-set. Contributor-privacy represents social network data as collected from individuals in a population by means of a survey, a realistic scenario for many social science applications. When survey data are collected in this fashion, it's likely that the data will not form a single consistent social network. There will likely be people in the population who do not submit a survey; there may be disagreement about the existence of edges (see Figure 4.1).



Name	Age	STD	Orientation	List Sexual Partners in Last 2 Years	List Friends on Campus
Alice	19	Yes	Bi	Bob, Dana, Jimmie	Dana, Eun...
Bob	20	No	Straight	Alice, Sarah...	Alice, Carl...
Carl	21	No	Gay	George, Frank	Eun, George
Dana	18	No	Straight	N/A	Alice
George	20	Yes	Straight	N/A	Carl, Eun

Fig. 4.1.: Unlike tabular data, participants in social network studies provide information about each other. This information may be incomplete or inconsistent.

Incomplete or inconsistent data may not negatively effect analysis utility. As we demonstrate below, a monolithic, consistent social network isn't requisite for useful analyses such as degree distributions or local clustering-coefficient distributions. In fact, imposing mutual friendship edges may produce a less accurate view of the network; some relationships are simply ambiguous or one-sided.

Additionally, this inherent ambiguity is an advantage for providing subject privacy. Consider 'subject data' to be nodes, edges or other information about the network which may be reported in survey participants' contributed information. Given a dataset and a specific set of subject data,d_s, we can examine the effects of the presence or absence of this data on the analysis results. Analogous to local function sensitivity, we define this impact as follows:

Definition 4.2 ***Subject Data Sensitivity*** **(**$\delta F(d_s)$**):** *Given a deterministic (non-privatized) function* $F : \mathcal{D} \rightarrow R^k = A$, *and a set of subject data* d_s, *the* ***Subject Data Sensitivity*** *of* $F(d_s)$ *is* [1]*:*

$$\delta F(d_s) = \max_{D_1, D_2} \|F(D_1) - F(D_2)\|_1$$

over all pairs of neighboring data-sets D_1, D_2 *such that* $D_1 = D_2/d_s$.

Because noise added to protect contributor data does not satisfy any explicit protection guarantee for subject data, we consider, as a lower bound on subject privacy, the context in which raw analysis results are published without the addition of any privatization noise. In Chapter 7 we introduce De Facto Privacy, which provides a formal metric for analyzing the degree to which a deterministic data analysis and publication scheme magnifies the impact of an attacker's uncertainty about the data. Below we provide a brief illustrative example of the fundamental concept.

When the results are published, $\delta F(d_s)$ is the only information the results provide an attacker about the existence of d_s in the network. However, due to the choice of publication schemes, the ambiguity in the collected survey data, and the attacker's own potential uncertainty about the collected network, the existence of d_s may not the only possible explanation for the effect $\delta F(d_s)$.

Consider a survey question: "How many people in the Purdue CS Department are you friends with?" Summing the data collected from this and dividing by two gives us an estimate of the number of friendships in the department, where one-sided edges and edges leading to non-surveyed individuals will be counted as half-edges. Define $\mathit{edge\text{-}count}(G) = |E|/2$.

Consider the friendship between Prof. Alice and Prof. Bob, $d_s = \mathit{edge}(A, B)$. If both individuals submit surveys and report on this friendship in their tally, they will collectively increase the total count by 2, ie $\delta F(d_s) = |\mathit{edge\text{-}count}(G) - \mathit{edge\text{-}count}(G/d_s)| = 2$.

[1] The L_1-norm of $x \in \Re^n$ is defined as $\|x\|_1 = \Sigma_{i=1}^n |x_i|$.

However, if instead Alice and Bob were each friends with two outside individuals ($Diane$ and Ed) who did *not* submit surveys, this would *also* increase the total count by 2. Note that $|\{edge(A, D) + edge(A, E) + edge(B, D) + edge(B, E)\}|/2 = 2 = \delta F(d_s)$, thus $edge\text{-}count(G) = edge\text{-}count(G + \{D, E\}/d_s)$.

Additionally, if non-participating individual Carla had remembered to submit her survey counting her four friends Alice, Bob, Fran and George (who had already submitted their surveys), that would *also* increase the total count by 2. Thus, $|\{edge(C, A) + edge(C, B) + edge(C, F) + edge(C, G)\}|/2 = 2$.

By itself, $\delta F(d_s)$ does nothing to indicate which of the many possible explanations is correct, thus offering some protection for the privacy of the subject data. Intuitively, the difficulty of ruling out alternate explanations, as described above, will cause analyses that possess low subject-sensitivity and are performed over large data-sets to impose a significant burden on an attacker attempting to use background knowledge to target a specific individual. A formal exploration of these concepts will be presented in Chapter 7 of this work.

Finally, we include a theoretical result regarding the identifiability of the network as whole. Given a analysis set A with total sensitivity ΔA, we note the following implication of the definition of contributor-privacy.

Theorem 4.3 *If R is the complete contributor-privatized results of analysis set A over true graph G, then there exists at least two networks G_1, G_2 such that $G_1 \neq G_2$ and $\frac{Prob[A(G_1)=R]}{Prob[A(G_2)=R]} \leq e^{2\epsilon}$. Thus, an attacker cannot use R to determine with certainty the true original graph G.*

Proof: *WLOG, we will assume that G_1 is the true original graph. We create G_2 by adding one 'leaf' node l which has a single edge connecting it to another, arbitrary, node a in G_1. Because l has only one neighbor, this new (l, a) edge is observed in the ego-networks of only two nodes: l and a. 1-Contributor-privacy protects data that is contributed by one node, ensuring that regardless of the analysis being performed, the guarantee in Definition 4.1 will hold. We want to protect the information contributed* *by two nodes, (2-contributor-privacy), which would cover the existence of the edge* (l, a).

Consider an intermediate data structure G_{1-2} *which contains a's observation of* l *but does not contain* l*'s ego-net information (possibly because* l *did not participate in the data-set themselves). Then we have that, under contributor-privacy,* G_1 *is a 1-step neighboring world of* G_{1-2} *and* G_{1-2} *is a 1-step neighboring world of* G_2*. So, for any* R*, we have:* $\frac{Prob[A(G_1)=R]}{Prob[A(G_{1-2})=R]} \leq e^{\epsilon}$ *and* $\frac{Prob[A(G_{1-2})=R]}{Prob[A(G_2)=R]} \leq e^{\epsilon}$.

Thus:

$$\frac{Prob[A(G_1)=R]}{Prob[A(G_{1-2})=R]} \times \frac{Prob[A(G_{1-2})=R]}{Prob[A(G_2)=R]} \leq (e^{\epsilon})^2$$

$$\frac{Prob[A(G_1)=R]}{Prob[A(G_2)=R]} \leq e^{2\epsilon}$$

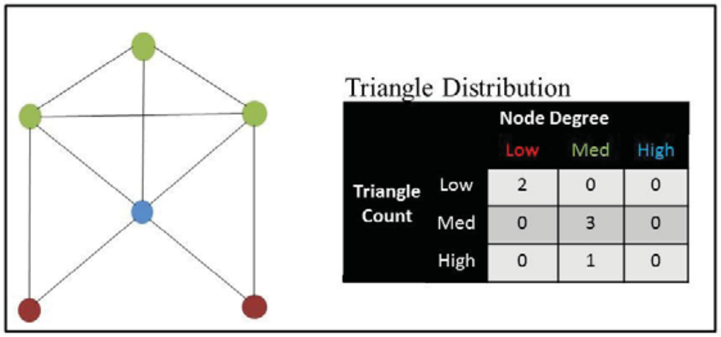
Note that because the choice of a *was arbitrary, there will be many possible instantiations of graph* G_2*, and thus for any given published privatized analysis set* R *there will be a (possibly quite large) pool of possible original networks* G.

Subject sensitivity is dependent on the choice of analysis, but the above protection, which is based on amounts of contributed information, holds regardless of the analysis set. In general under contributor-privacy, network features which fall within many individual's ego-networks (and thus are directly observable by many individuals) may be visible in the privatized results, depending on the choice of analysis. Thus, as with edge-privacy, the overall effect of high-degree nodes in the network may not be protected. However, less observed nodes are guaranteed more protection, and choosing analyses with lower subject-sensitivity will provide better protection for higher-degree nodes (we will discuss De Facto protection in Chapter 7). In the remainder of this chapter, we will demonstrate a number of contexts and analyses for which contributor-privacy protection is appropriate. For more sensitive contexts, partition-privacy (if applicable) can provide very strong protection while still enabling high-utility analysis. Partition-privacy is introduced in Chapter 5.

4.2 Basic Algorithms

In this section we will present several easy-to-use algorithmic tools that can enable social network researchers to learn about populations while guaranteeing contributor-privacy. In the next section we will demonstrate the practical application of contributor-private analysis on a diverse set of real world social network data-sets.

4.2.1 Subgraph Counts



Triangle Distribution

		Node Degree		
		Low	Med	High
Triangle Count	Low	2	0	0
	Med	0	3	0
	High	0	1	0

Fig. 4.2.: The triangle distribution allows us to present clustering information with an contributor-sensitivity of 1.

We propose a method for privatizing information about triangle counts and clustering coefficients under contributor-privacy, using a modified version of the query that more closely mimics the information gathered from a social network survey. To do this, we introduce a simple, powerful method that can be applied to gather privatized estimates of a variety of useful statistics over nodes in the graph.

By focusing on protecting the knowledge each individual has about their role with respect to the network, contributor-privacy fits naturally with the techniques of *ego-network analysis*, an approach to social network analysis that considers the network as viewed by the individuals belonging to it [54]. In ego-network analysis, a network with n members is broken into n overlapping ego-network subgraphs, each consisting of a individual ‘ego’ node and his or her immediate neighborhood of friends (referred to as alters). Algorithm 1 presents survey collecting information about the triangles in an individual's ego-network.

Algorithm 1 A survey gathering information about triangles.

```
function TriangleQuery
    $friendlist$ ← Query("Who are your friends?")
    $friendpairs$ ← CrossProduct($friendlist, friendlist$)
    $outdegree$ ← Size($friendlist$)

    $triangles$ ← Query("Which of these pairs are friends with each other?", $friendpairs$)
    $trianglecount$ ← Size($triangles$)
    return ($outdegree, trianglecount$)
end function
```

The only data that is retained by the researcher is, for each individual x: *outdegree*(x), the number of friends the individual has, and *trianglecount*(x), the number of triangles the individual participates in. These statistics are sufficient to determine the local clustering co-efficient of the node: the ratio between the number of triangles the node participates in and the maximum possible number of triangles for a node of that degree [31].

Out-degree and local clustering data from this survey can be collected into a two-dimensional histogram that provides detailed information about the patterns of social cohesion of the graph and has a very low sensitivity under contributor-privacy (see Figure 4.2): removing or adding an individual's survey data to the histogram only alters one partition count by at most one, and thus the noise required to privatize this data-structure would be very small. Histograms with fewer partitions and larger count values in each partition are less sensitive to added noise; we propose Algorithm 2 that produces a very flexible, robust, and safely privatized representation of the social cohesion patterns in the network using local triangle counts.

Algorithm 2 takes as input two node-degree threshold values, deg_{low}, deg_{med} and uses these to partition the $(outdegree, trianglecount)$ data-points collected from the

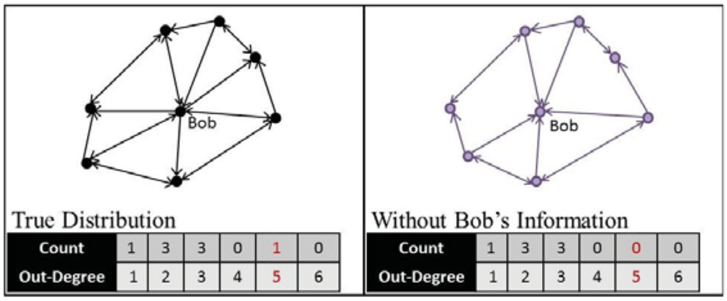
Algorithm 2 Privatizing local clustering coefficient distribution data.

```
function PRIVATECLUSTERING($deg_{low}, deg_{med}, data$)
    Initialize($bins[][]$)
    for all $(nodeDegree, triangleCount) \in data$ do
        $degBin \leftarrow$Partition($nodeDegree, deg_{low}, deg_{med}$)
        $localCluster \leftarrow triangleCount/(nodeDegree * (nodeDegree - 1))$
        $triBin \leftarrow$Partition($localCluster, 1/3, 2/3$)
        $bin[degBin][triBin] \leftarrow bin[degBin][triBin] + 1$
    end for
    for $i = 0 \rightarrow 2, j = 0 \rightarrow 2$ do
        $bins[i][j] \leftarrow bins[i][j]+$ LaplacianNoise(1)
    end for
    return $bins$
end function
```

TriangleQuery survey into low, medium and high degree nodes. The algorithm then computes the local clustering coefficient of each node and further partitions nodes by these values, creating a histogram with nine partitions (see Figure 4.2). Laplacian noise sufficient to cover a function sensitivity of 1 is added to each partition, and the privatized result may be released. We can consider the effect of this noise in terms of how many of the noisy, privatized partition counts can be expected to differ measurably from their true values. With only nine counts and a sensitivity of 1, the expected number of privatized partition counts that will differ from their true values by more than 3 is less than 0.25. The released histogram accurately captures useful information about the distribution of local patterns across the graph.

The same approach can be used to collect and privatize any information available within an ego-network by restructuring the survey as needed. For example, replacing question 2 in Algorithm 1 by the question "For each of your friends, add a check mark if the two of you share at least one additional, mutual friend" will collect information about the probability that an edge participates in a triangle. The question "Are you part of a group of at least k friends who are all mutual friends with each other?" collects statistics about cliques in the graph.

4.2.2 Degree Distribution



True Distribution

Count	1	3	3	0	1	0
Out-Degree	1	2	3	4	5	6

Without Bob's Information

Count	1	3	3	0	0	0
Out-Degree	1	2	3	4	5	6

Fig. 4.3.: Contributor sensitivity = 1. Protecting the out-edges of a node provides privacy with relatively little effect on the degree distribution.

Although edge-privacy requires comparatively little noise to protect individuals in degree-distribution analyses (see Figure 3.4), contributor-privacy requires even less noise. Here, we consider just the distribution of out-degrees, the result of asking participants, "How many friends do you have?" Removing one node and its out-links from the graph affects only one value in the degree distribution (Figure 4.3). Under contributor-privacy, a high-degree node may still leave evidence of its presence in the data-set through the out-degrees of its friends. However, this may not significantly compromise subject privacy. The set of possible explanations for a slightly higher-than-expected degree among nodes in the graph is large: they may represent additional friendships among the nodes, or outside friendships with individuals who were non-participants in the survey. Exploiting this vulnerability to guess the presence of a high-degree node with any certainty would require an attacker to possess extensive information about the true social network and survey participation. We will explore this concept in more detail in Chapter 7.

4.2.3 Edge Properties

We now propose a method for collecting privatized information about a class of network statistics that can be characterized as propositions on network edges. Assume we are given a node n with edge-set $E_n = \{(n,a)|a \in neighborhood(n)\}$, node labels n_l and $\{a_l|a \in neighborhood(n)\}$, and edge labels $\{e_l|e \in E_n\}$. Within this framework, we can define propositions that capture useful information about individual edges.

Edge Propositions $(e \in E_n)$

- **Mutual(e):** $P_{mutual}((n,b)) = [(b,n) \in E_b]$
- **MutSigned(e):** $P_{mutsigned}((n,b)) = [((b,n) \in E_b) \ and \ sign(n,b) = sign(b,n)]$
- **Friend-Type(e):** For example, $P_{Female}(n,b) = [Female?(b)]$
- **Same-Type(e):** $P_{same-type}(n,b) = [n_l = b_l]$

In this section we describe how such edge information can be aggregated at the node level, then contributed to aggregate network statistics, and privatized under contributor-privacy as a distribution with low sensitivity. Individuals can be expected to be familiar with basic information about their relationships and their friends. Algorithm 3 describes a general approach to collecting edge information from an individual that may be implemented with any of the edge propositions listed above. EdgePropertyQuery(Mutual()) asks individuals for the proportion of their network edges that are mutual (for instance, whether someone they follow is also one of their own followers); this collects information about the node's reciprocity. EdgePropertyQuery(SameType()) asks individuals for the proportion of their friends are similar to them (for example, whether their friends share their race); this collects information about the node's degree of homophily (see Section 2.3.2).

Algorithm 3 A survey gathering information about the properties of friendship edges.

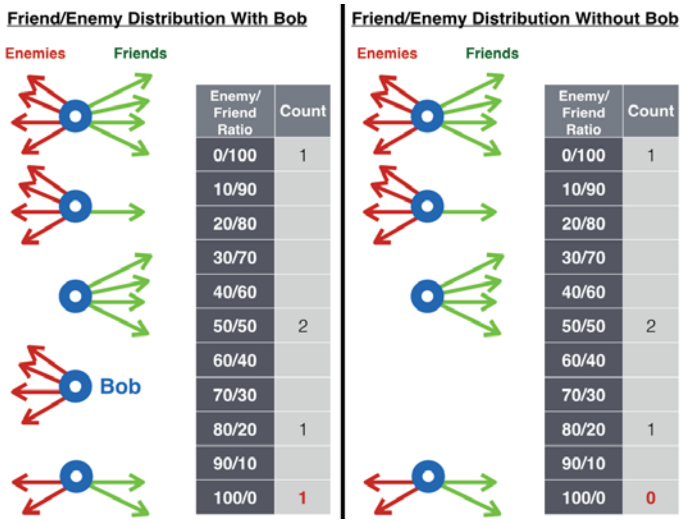
```
function EdgePropertyQuery (PROPERTY)
    $outdegree \leftarrow$ Query(“How many friends do you have within the PoI?”)
    $positivecount \leftarrow$ Query(“How many of your friendships have PROPERTY?”)
    return $(outdegree, positivecount)$
end function
```

Once this information is collected at the node level, it can be aggregated into a distribution across the network (see Figure 4.4). Contributor-privacy requires that each node's contributed information be protected. Because the information is collected into a histogram, as in the degree distribution, the contributor-privacy sensitivity is 1 and the distribution can be privatized with relatively little noise. The complete procedure for creating and privatizing the distribution is given in Algorithm 4.

Privatized edge-property distributions can be applied to a variety of social network analysis contexts. For example, data collected from EdgePropertyQuery(SameType$^{\text{Grades}}$()) would provide information about the extent to which students associate

Algorithm 4 Privatizing edge property distribution data.

```
function PrivateEdgePropertyDistribution($(data, precision)$)
    Initialize ($bins = [] * (10^{precision} + 1)$)
    $binLabels \leftarrow [”0/10^{precision}”, ”1/(10^{precision} - 1)”, ...”10^{precision}/0”]$
    for all $(outDegree, positiveCount) \in data$ do
        $ratio \leftarrow positiveCount/outDegree$
        $binNum \leftarrow round(10^{precision} * (ratio))$
        $bins[binNum] \leftarrow bins[binNum] + 1$
    end for
    for $i = 0 \rightarrow 10^{precision} + 1$ do
        $bins[i] \leftarrow bins[i] +$ LaplacianNoise(1)
    end for
    return $bins, binLabels$
end function
```


Friend/Enemy Distribution With Bob
Enemies
Friends
Bob
Friend/Enemy Distribution Without Bob
Enemies
Friends

Enemy/ Friend Ratio	Count
0/100	1
10/90	
20/80	
30/70	
40/60	
50/50	2
60/40	
70/30	
80/20	1
90/10	
100/0	1

Enemy/ Friend Ratio	Count
0/100	1
10/90	
20/80	
30/70	
40/60	
50/50	2
60/40	
70/30	
80/20	1
90/10	
100/0	0

Fig. 4.4.: Contributor sensitivity = 1. Information about prevalence of edge properties such as reciprocity, homophily, or prevalence of particular edge labels (such as friend/enemy labels), can be collected from nodes with low sensitivity.

with others at the same level of academic proficiency. The data collected from EdgePropertyQuery (Mutual-Signed()) would provide information about the extent to which positive and negative edges are reciprocated in kind (see Section 4.4.2).

4.2.4 Centrality: Popularity Graph

We propose a very different approach for collecting and privatizing information about influential nodes within a network; one that satisfies contributor-privacy (by protecting individuals' data contributions) and leverages individuals' knowledge about their community. Recall that viable centrality analyses beyond degree-distributions do not currently exist for the edge-privacy or node-privacy standards. We define a *popularity graph*: a synthetic network that represents the social structure among influential community members (Algorithm 5).

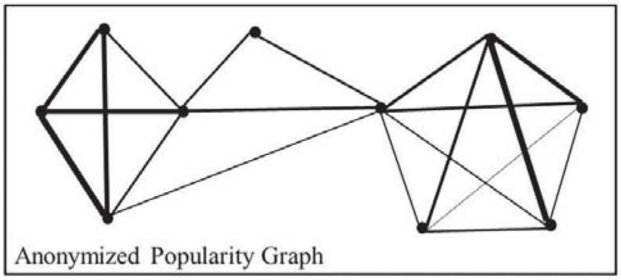


Fig. 4.5.: A Popularity Graph with edge thickness indicating edge-weight

Given a population of interest in which the number of individuals in the total population set is public information, [2], the algorithm proceeds as follows. Individuals in the population are asked to: "list your three most popular friends within the specified population group". The algorithm proceeds in two steps: a data aggregation and privatization step, followed by a post-processing step which uses the privatized data to produce the popularity graph. The sensitivity of the first step is 3, while the second step manipulates only privatized data and thus incurs no sensitivity cost.

The first step in the algorithm builds a social network across as follows: A base graph is created containing sufficient nodes for all members of the population of interest, and undirected edges of weight 0 are added between all pairs of nodes. The data collected from the survey is then added to the graph: when two popular people are listed on the same survey, the weight of the edge connecting them is incremented. Thus, each individual's data increments the weight of the three edges connecting the three 'popular' nodes contributed by that individual. The sensitivity of the popularity graph is 3, since a maximum 3 edge-weight values can change if a participant adds or retracts their data.

To privatize the data, appropriate Laplacian noise to cover a function sensitivity of 3 is added to all edge-weights. Then the post-processing step is applied: edges with low weight are eliminated, and the graph is anonymized. The resulting weighted

[2] Note that this is *not* the number of individuals who contribute their information to the data-set, which is protected under contributor-privacy

Algorithm 5 Privatizing centrality data.

```
function PrivateCentrality($(importanceT, data_I)$)
    $V \leftarrow N$
    $E[i][j] \leftarrow 0 \; \forall i, j \in V$
    for all $i \in I$ do
        $\forall p_j, p_k \in data_I[i]$, $E[p_j, p_k] \leftarrow E[p_j, p_k] + 1$
    end for
    for all $i, j \in popular - population$ do
        $E[i, j] \leftarrow E[i, j] +$ LaplacianNoise(3)
        if $E[i, j] < importanceT$ then
            $E[i, j] \leftarrow 0$
        end if
    end for
    return $PopularityGraph = (V, E)$
end function
```

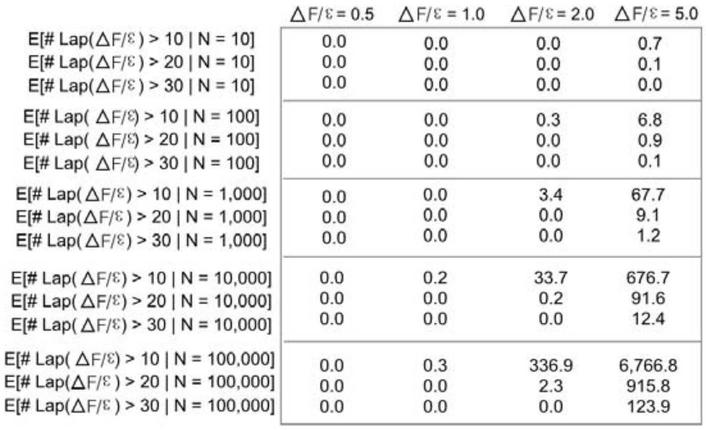
popularity graph is published (Figure 4.5). This graph can be used to understand the underlying social influence structure of the population, identifying social clusters and the bridges between them. The privacy of data provided by the query participants is fully protected; however, the subjects who appear as popular nodes in the graph will clearly be less secure and this analysis may not be appropriate in all contexts (in a sexual relationship network, for example, analyses with greater subject privacy would be preferable). However, for many populations though, the popularity graph should be sufficient protection: anonymity, noisy edges, and the fact that the artificially-constructed graph will lack detailed substructures often used for re-identification attacks, will all contribute to protecting the privacy of the query subjects.

4.3 Utility Analysis

The basic contributor-privacy algorithms we have presented above have low sensitivity: Triangle counts, degree distributions, edge-property distributions each have a sensitivity of 1, while popularity graphs have a sensitivity of 3. In this section we'll discuss the relationship between sensitivity and utility.

Recall from section 2.1 that the sensitivity of a set of analyses over a single data-set is equal to the sum of their individual sensitivities. Furthermore, if an analysis is performed over two disjoint data-sets such that an individual can contribute to at most one set of analysis results, the sensitivity of the two analyses is computed independently. A standard social network analysis scenario involves performing a set of analyses across several networks and then comparing the results; for example, we might perform a degree-distribution and two edge-property distribution analyses across four different networks. The sensitivity of the analysis for each network is equal to 3 (a sensitivity cost of 1 for the degree-distribution, and a total cost of 2 for the two edge-property distributions). In order to privatize this analysis set, laplacian noise sufficient to privatize a sensitivity of 3 must be added to each analysis (the degree-distribution and both edge-property distributions) across all four networks.

The effect of this noise on analysis results is dependent on two factors: the size of the data-set and the size of the output structure being privatized (the number of noise samples taken). Because privatization noise must be added to every 'bucket' of histogram-formatted data, privatized data-structures with more bucket counts have a higher probability of sampling at least one large value of noise: A LCC distribution with 10 counts has a very low probability of sampling a large noise value during privatization, in contrast to a Popularity Graph with 500,000 edge weights. Whether a sampled noise value is sufficiently large to obscure the true data patterns is dependent on the size of the data-set: In a data-set with 100 individuals, a sampled noise value of 10 may be large enough to overwhelm the true value of a histogram count. In a data-set with 10,000 individuals, the effect of a noise value of 10 on the aggregate data patterns is less significant. Figure 4.6 gives the expected number of large noise values sampled depending on the output data-structure size and analysis sensitivity (with $\epsilon = ln(2)$).



	$\Delta F/\varepsilon = 0.5$	$\Delta F/\varepsilon = 1.0$	$\Delta F/\varepsilon = 2.0$	$\Delta F/\varepsilon = 5.0$
$E[\# Lap(\Delta F/\varepsilon) > 10 \mid N = 10]$	0.0	0.0	0.0	0.7
$E[\# Lap(\Delta F/\varepsilon) > 20 \mid N = 10]$	0.0	0.0	0.0	0.1
$E[\# Lap(\Delta F/\varepsilon) > 30 \mid N = 10]$	0.0	0.0	0.0	0.0
$E[\# Lap(\Delta F/\varepsilon) > 10 \mid N = 100]$	0.0	0.0	0.3	6.8
$E[\# Lap(\Delta F/\varepsilon) > 20 \mid N = 100]$	0.0	0.0	0.0	0.9
$E[\# Lap(\Delta F/\varepsilon) > 30 \mid N = 100]$	0.0	0.0	0.0	0.1
$E[\# Lap(\Delta F/\varepsilon) > 10 \mid N = 1,000]$	0.0	0.0	3.4	67.7
$E[\# Lap(\Delta F/\varepsilon) > 20 \mid N = 1,000]$	0.0	0.0	0.0	9.1
$E[\# Lap(\Delta F/\varepsilon) > 30 \mid N = 1,000]$	0.0	0.0	0.0	1.2
$E[\# Lap(\Delta F/\varepsilon) > 10 \mid N = 10,000]$	0.0	0.2	33.7	676.7
$E[\# Lap(\Delta F/\varepsilon) > 20 \mid N = 10,000]$	0.0	0.0	0.2	91.6
$E[\# Lap(\Delta F/\varepsilon) > 30 \mid N = 10,000]$	0.0	0.0	0.0	12.4
$E[\# Lap(\Delta F/\varepsilon) > 10 \mid N = 100,000]$	0.0	0.3	336.9	6,766.8
$E[\# Lap(\Delta F/\varepsilon) > 20 \mid N = 100,000]$	0.0	0.0	2.3	915.8
$E[\# Lap(\Delta F/\varepsilon) > 30 \mid N = 100,000]$	0.0	0.0	0.0	123.9

Fig. 4.6.: Expected number of high noise values given function sensitivity and number of noise samples taken, with $\epsilon = ln(2)$.

Our hypothetical analysis set above (assuming a degree-distribution cut-off of 80 and an edge-property range of 11) has a total of $4 * (80 + 22) = 408$ histogram buckets across the four network analyses, and a total sensitivity of 3. To compute the expected number of large noise values, we use the cumulative probability distribution of the laplacian distribution, given in Lemma 4.4. In our example, the expected number of noise values larger than 10 appearing across the entire four network analysis set is 19.84, and the expected number of noise values larger than 35 is 0.06. In real world networks with tens of thousands of nodes, such as the ones we investigate in the next section, this amount of noise is unlikely to have a significant impact on results utility. We will demonstrate this empirically. In much smaller networks, it may be necessary to reduce the analysis output space (for example, a researcher may choose a smaller degree-distribution cut-off in smaller networks), increase ϵ (and accept the reduced level of privacy protection), or perform fewer analyses.

Lemma 4.4 ***Expected Large Noise Values:*** *Given an analysis with sensitivity ΔF and an output of size H (requiring H noise samples to privatize), a privacy parameter ϵ, the expected number of Laplacian noise values sampled larger than k is:*

$$E[\#Lap(\frac{\Delta F}{\epsilon}) > k|H] = H * 0.5e^{-k/\frac{\Delta F}{\epsilon}}$$

4.4 Practical Application

The contributor-private analysis techniques described in this chapter are carefully designed to minimize analysis sensitivity and output size, such that privacy can be achieved with relatively little added noise. We believe that these algorithms present an effective tool-set for easily implemented, high-utility, privacy-preserving social network analysis.

As evidence of this, we now give a in-depth practical demonstration of contributor-private network analysis techniques on real world social networks. For our experiments we use the networks provided by an widely-used social network analysis resource: the Stanford Large Network Dataset Collection (SNAP) [55]. These networks have been published online as anonymized edge and node sets, and have been referenced by many researchers working across the social network analysis field. We will thus assume that any privacy risks inherent in the simply-annonymized data predate this thesis due to their public availability, and we will include both privatized and non-privatized analysis results for comparison purposes.

To demonstrate the range of contributor-private analyses, we perform three analysis sets across three different categories of networks. In the first two categories, we compare analysis results across pairs of networks that possess the same data-structure format but are drawn from very different populations. In the third category, we explore a challenging privacy scenario by demonstrating a higher sensitivity analysis against a smaller data-set with a large output space:

- **Directed Networks:** Enron Email Network, WikiVote Network
- **Signed Directed Networks:** Slashdot Zoo Enemy/Friend Network, Epinions Trust Network
- **Small Undirected Network:** Facebook Ego-Network

Recall from Section 4.3 that sensitivity is computed across the entire analysis set for each graph. In our first two analysis sets, both the total analysis sensitivity and the data-output size are relatively small in comparison to the size of the data, resulting in a very small effect on utility that is generally not visible when the analysis results are graphed. For these analyses we also include truncated graphs with reduced ranges such that the privatization noise is visible, as well as a record of the noise values added to each distribution. In the third analysis set the data-set is much smaller and our output space is much larger; thus, the effect of noise is more significant. In all other analyses in this dissertation, we use a default parameter of $\epsilon = ln(2)$–Note that, given Definition 2.2, a choice of $\epsilon = ln(2)$ provides the privacy guarantee that no result will be more than twice as likely to be produced by one data-set as by its neighbor. In the third analysis set in this section, we will demonstrate the effect of an increased ϵ on privatized results. Increasing ϵ weakens the privacy guarantee, but increases the utility of privatized results.

To compare results between networks of different sizes, we first normalize the distributions. This is a post-processing step that occurs after privatization and incurs no sensitivity cost. We use a natural definition for the privatized normalized distribution, as follows:

Definition 4.5 ***Privatized Normalization:*** *Given a distribution H_{priv} with privatized (noisy) counts: $p_1, p_2 ... p_k$, we define the* ***privatized normalization*** *as:*

$$H_{priv-norm} = p_1/n_{priv}, p_1/n_{priv}..p_k/n_{priv}$$

With $n_p riv = \Sigma_i p_i$.

4.4.1 Directed Networks: Voting and Email Data

In our first analysis set, we investigate two directed networks.

Data-sets

Our first data-set is the Enron Email network. During the Federal Energy Regulatory Commission's investigation of the company, about a half million of the company's internal emails were posted publicly online. A social network data-set has been drawn from this data by including nodes for each individual and adding directed edges to represent email exchanges (ie, an edge is added leading from node i to node j if individual i sent at least one email to individual j). The resulting graph has 36,692 nodes and 183,831 edges (ie, an average degree of 5). In this graph, contributor data consists of the emails sent by one individual, along with email relationships between individuals in the contributor's neighborhood (presumed to be potentially observable by the individual).

Our second data-set is taken from a procedure that occurs in Wikipedia's editor community. Content on Wikipedia is composed, edited and monitored by volunteers, and a subset of especially dedicated volunteers are given 'administrator' status with a higher level of access privileges. Administrator positions are awarded as the result of a community deliberation process that involves a voting round: fellow volunteers (both common volunteers and current administrators) vote on administrator candidates. A social network is drawn from this data by including nodes for every voter and candidate, and adding directed edges to indicate votes (ie, an edge is added leading from node i to node j if individual i voted in support of individual j). Note that this is not a bipartite graph: candidates can and often do cast votes for other candidates. The resulting graph has 7,115 nodes and 103,689 edges (an average degree of 14.5). In this graph, contributor data consists of all the votes cast by one individual, along with votes exchanged between individuals in the contributor's neighborhood (presumed to be potentially observable by the individual).

In analyzing these graphs, we will investigate the following questions:

1. Does the email network or voting network have higher transitivity? Transitivity in the voting network indicates groups of individuals mutually supporting each other's bids for administratorship. Transitivity in the email network may indicate collaboration networks among groups of coworkers.

2. Does the email network or voting network have a more hierarchical structure (many low and few high degree nodes)? Low degree nodes in the voting network indicate individuals who cast few votes, while low degree nodes in the email network indicate individuals who sent emails to only a few coworkers.

Privatized Analysis

For this analysis set we will collect a local clustering coefficient (LCC) distribution and a degree distribution from each network, and then privatize the results using laplacian noise, as described in the previous section. Because we are interested in the distribution of LCC values across the network as a whole, rather than binned by degree, for this experiment we use a one-dimensional LCC histogram in place of the two-dimensional histogram presented in Figure 4.2.

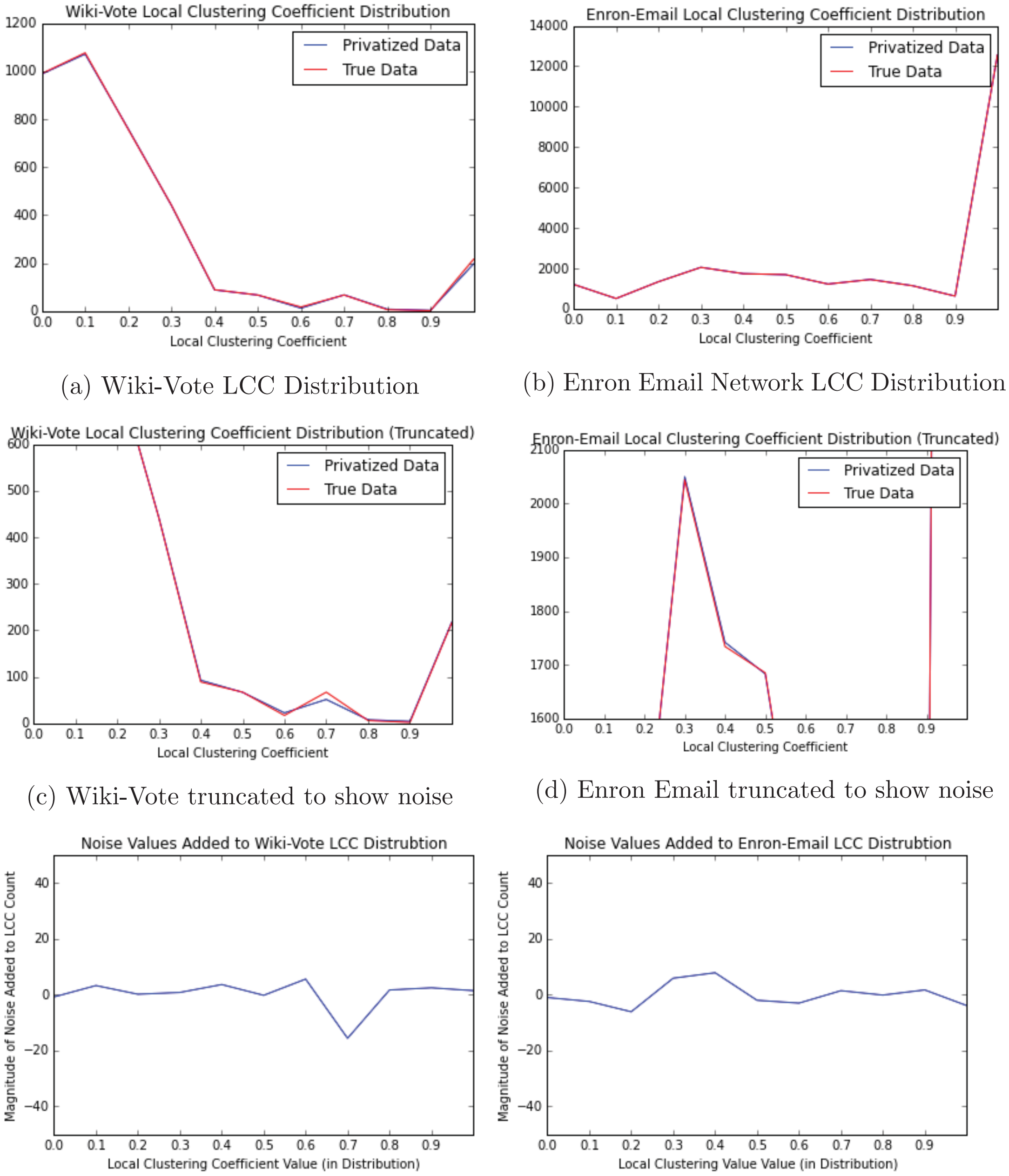
The total sensitivity of this analysis set is 2, with each analysis incurring a sensitivity cost of 1. Laplacian noise sufficient to cover this sensitivity is added to each output count in each analysis to produce the privatized results. The following two figures, Figure 4.7 and 4.8, display the output of the analysis set. Because the of the low sensitivity of these analyses, the privatization noise is very small in comparison to the scale of the data; we include plots with truncated axes in which the effect of the noise is visible. In general, the privatized results in this analysis set provide, literally, no visible loss in utility.

Privatized Results Comparison

To perform a privacy-preserving comparative analysis of two networks, we normalize the privatized results according to Definition 4.5. The resulting plots are presented in Figure 4.9 and 4.10.

To demonstrate that privatized social network analysis can provide useful insights into the populations and social dynamics being studied, we now return to the two questions we proposed above.

First, does the email network or the voting network show higher transitivity? Looking at the privatized LCC distribution data across both networks (see Figure 4.9) we see that most individuals in the network have a low clustering coefficient, while many individuals in the email network have a very high clustering coefficient: groups of individuals in the Wikipedia administrator candidate pool were less likely to trade votes amongst each other than employees at Enron were likely to collaborate in clique-like groups through email exchanges with their coworkers. This seems reasonably intuitive. Interestingly, we note that the Enron distribution also has some weight at a relatively lower LCC range of 0.2-0.6; potentially this represents individuals



(a) Wiki-Vote LCC Distribution

(b) Enron Email Network LCC Distribution

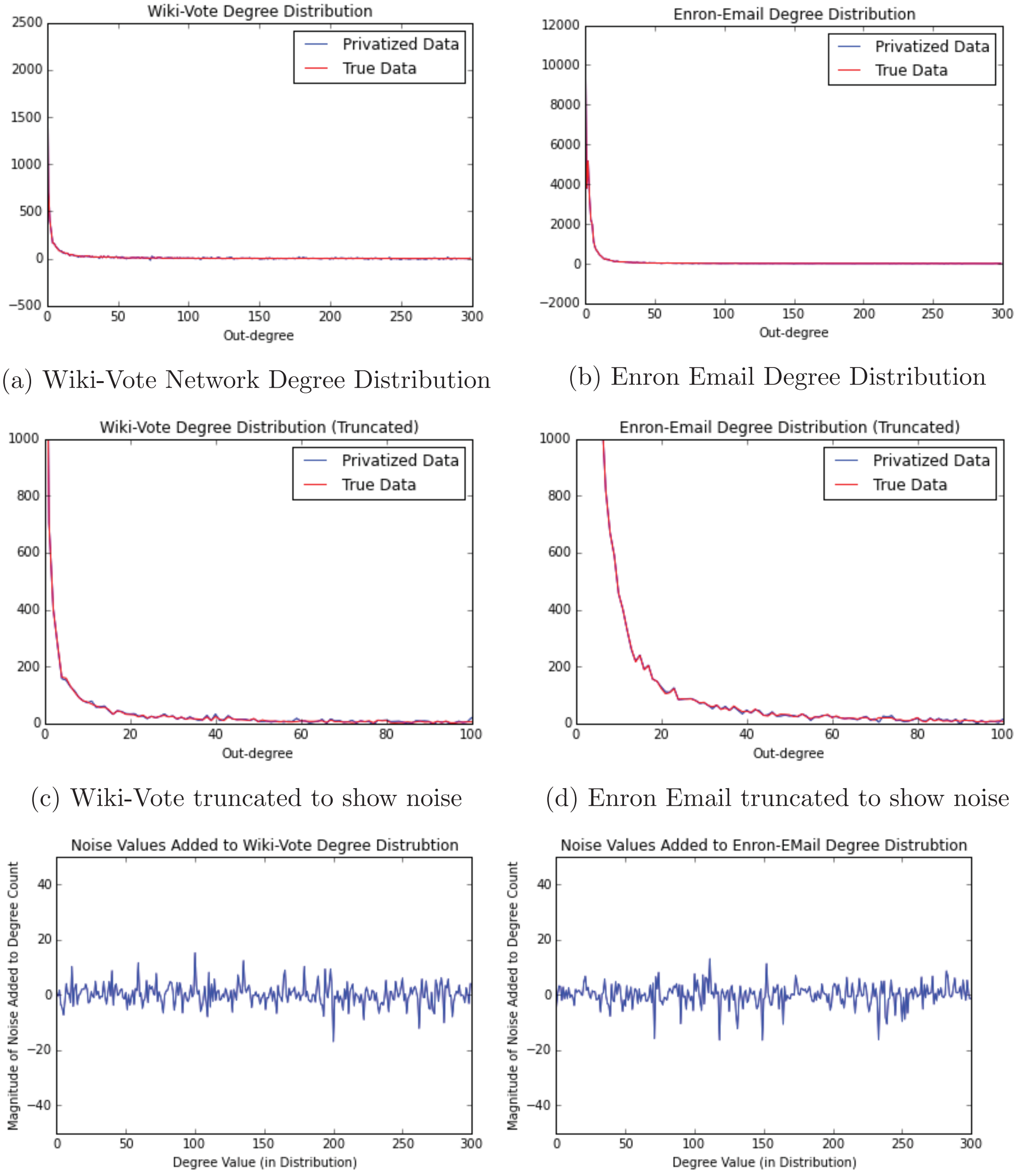
(c) Wiki-Vote truncated to show noise

(d) Enron Email truncated to show noise

(e) Wiki-Vote privatization noise

(f) Enron Email privatization noise

Fig. 4.7.: Local Clustering Coefficient (LCC) distribution data for the Enron Email and Wiki-Vote networks, privatized under contributor-privacy.



(a) Wiki-Vote Network Degree Distribution

(b) Enron Email Degree Distribution

(c) Wiki-Vote truncated to show noise

(d) Enron Email truncated to show noise

(e) Wiki-Vote privatization noise

(f) Enron Email privatization noise

Fig. 4.8.: Degree Distribution Data for the Enron Email and Wiki-Vote networks, privatized under contributor-privacy

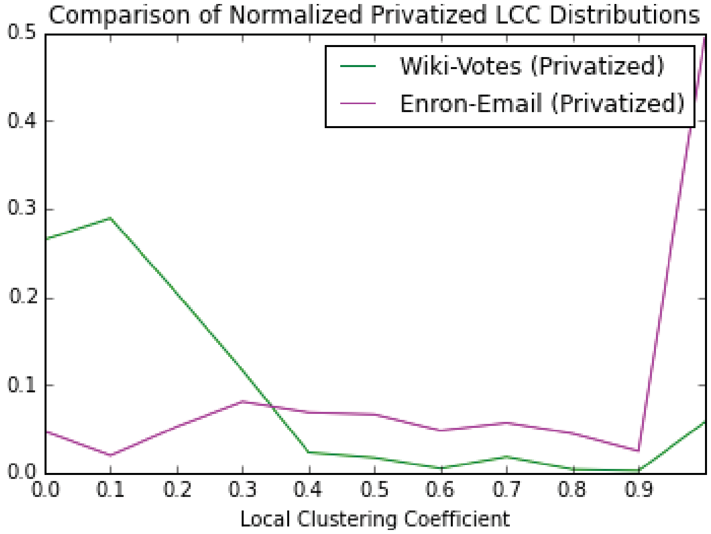


Fig. 4.9.: Comparison of normalized privatized local clustering coefficient distribution between Wiki-Vote and Enron Email networks.

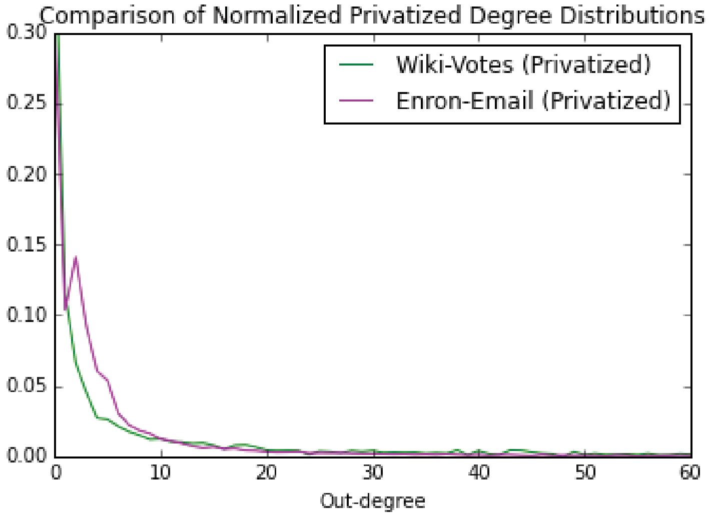


Fig. 4.10.: Comparison of normalized privatized degree distribution between Wiki-Vote and Enron Email networks.

in administrative staff positions or leadership positions who distributed email across disparate organizational groups that were not otherwise well-connected. We also note the small increase at LCC = 1 in the voting distribution; this suggests that some portion of administrator candidates did form voting cliques to mutually support each other's bids for administratorship.

Next, we investigate whether the email network or voting network showed a more hierarchical degree distribution (see Figure 4.10) . Privatization noise and the degree cut-off of 60 reduces the level of detail available about the few individuals in each graph with very high degree (this is a natural consequence of protecting these distinctive individuals' privacy). However, looking at trends across both populations as a whole, we see the distribution curves are very similar for both for very low degree nodes and higher degree nodes. We also note, though, that the email network presents a distinctive behavior around degree 3: There are many nodes in the Enron graph that have only one or two email partners, and there is a another large set of nodes that has between four and eight email partners. This may be evidence of collaboration substructures specific to the company organization. If node labels specifying position type were added to the network data, we could explore this question further in a two-dimensional privatized histogram (similar to the histogram recording degree and LCC presented in Figure 4.2).

4.4.2 Signed Networks: Friend/Enemy and Trust Graphs

In our second analyses set, we move onto two directed networks with signed edges that are labeled as either positive or negative.

Data-set

Epinions is a consumer review site that allows users to report trust (or distrust) relationships with other reviewers. This information is combined into a single network referred to as the Web of Trust, which is then used to algorithmically determine which reviews are displayed to a user in the future. A signed, directed social network is drawn from this data by including a node for each user and a signed directed edge for every trust relationship (ie, when user i registers a trust relationship with user j, a directed edge from i to j is added with label +1; for distrust relationships a −1 label is used). The complete network has 75,879 nodes and 508,837 edges, with an average degree of 6.7. Contributor data in this network is the set of trust and distrust relationships submitted by the user, and the set of trust and distrust relationships as received and observed by the user.

Slashdot is a technology-related news aggregater and forum with a very active community. The Slashdot Zoo, introduced in 2002, allows users to tag other users as either "friend" or "foe". A signed, directed network is drawn from this data by including a node for every user who participates in the Zoo, along with directed signed edges indicating friend and foe tags (when user i tags user j as a "foe", a directed edge with label −1 is added from i to j; friendship edges are labeled +1). The complete network has 82,168 nodes and 948,464 edges, with an average degree of 11.5. Contributor data in this network is the set of friend and foe tags registered by the user, and the set of friend and foe tags as received and observed by the user.

In this analysis set, we'll investigate the following questions:

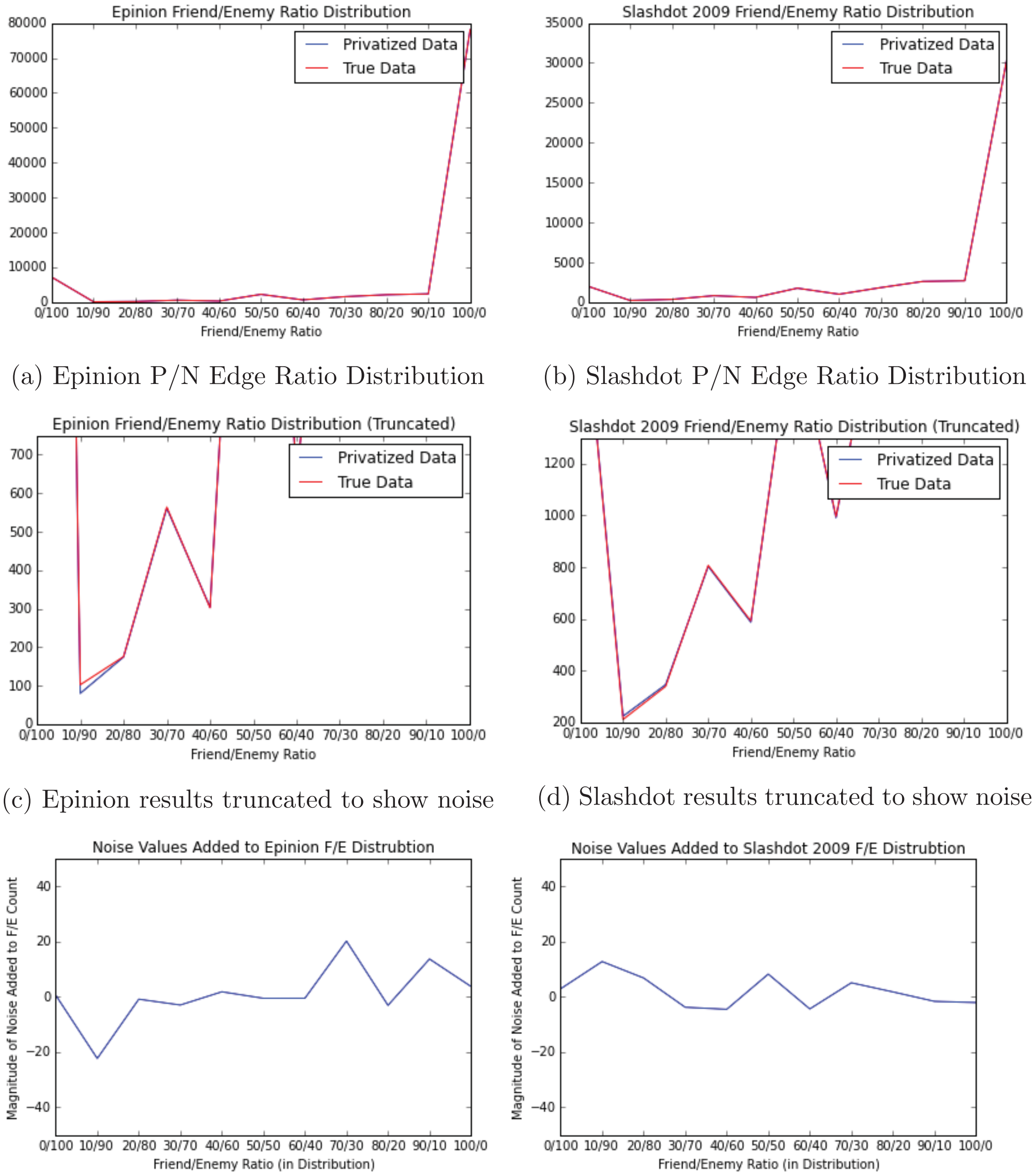
1. In signed networks, individuals' outdegree can be broken into counts of positive and negative edges. If the individual participates more in negative relationships than positive ones, their negative outdegree will be greater relative to their positive outdegree. Do individuals tend to be more negative in the social Zoo network where 'foe' relationships may not be intended seriously? Or do they tend to be more negative in the trust network, where negative edges have a serious meaning and actual consequences on future interactions?

2. Are individuals more likely to receive responses (form mutual relationships, regardless of sign), to the social relationships extended in the Zoo network or the trust relationships extended in the Epinions network?

3. Considering sign, are individuals more likely to recieve responses in kind to relationship edges extended in the Zoo network or the trust network? How likely are people to indulge in mutual friendships and foe-ships in each network?

For this analysis set we will collect four types of edge-property information across the two networks. We collect two edge property ratio distributions: the first counting the ratio of positive to negative edges that a node participates in, and the second counting the ratio of mutual edges (regardless of sign) that a node participates in. Each of these distributions has a sensitivity of 1. We also collect statistics about individuals' tendency to have their extended edges reciprocated in kind: we collect counts of the number of individuals who received positive responses to more than half of their extended positive edges (and the complementary count of those who did not), as well as a count of the number of individuals who received negative responses to more than half of their extended negative edges (and the complementary count of those who did not). Each pair of complementary counts can be represented as a two-bin histogram, which we then privatized and normalized to attain privatized estimates of the probability that a random node received in-kind responses to a majority of its positive and negative edges. Note that the same individual may contribute to both count pairs (eg, an individual whose positive edges are all reciprocated but whose negative edges are not reciprocated will contribute to the primary count of the positive count pair and the complementary count of the negative count pair). Thus, these analyses incur a sensitivity cost of 2, and the sensitivity of the full analysis set is 4. Laplacian noise sufficient to cover this sensitivity is added to each output count in each analysis to produce the privatized results.

The following three figures, Figure 4.11, Figure 4.12 and Figure 4.4.2, display the output of the analysis set. Because the privatization noise is very small in comparison to the scale of the data and the analyses have been carefully designed to incur minimal sensitivity cost, the effect of privatization noise on the results is again negligible. In the first two analyses, we include plots with truncated axes in which the effect of the noise is visible, and in the third figure we extend the precision to three decimal place in order to ensure the effect of added noise is clear. In general, we can see that the privatization noise in this second analysis set also has negligible effect on utility.



(a) Epinion P/N Edge Ratio Distribution

(b) Slashdot P/N Edge Ratio Distribution

(c) Epinion results truncated to show noise

(d) Slashdot results truncated to show noise

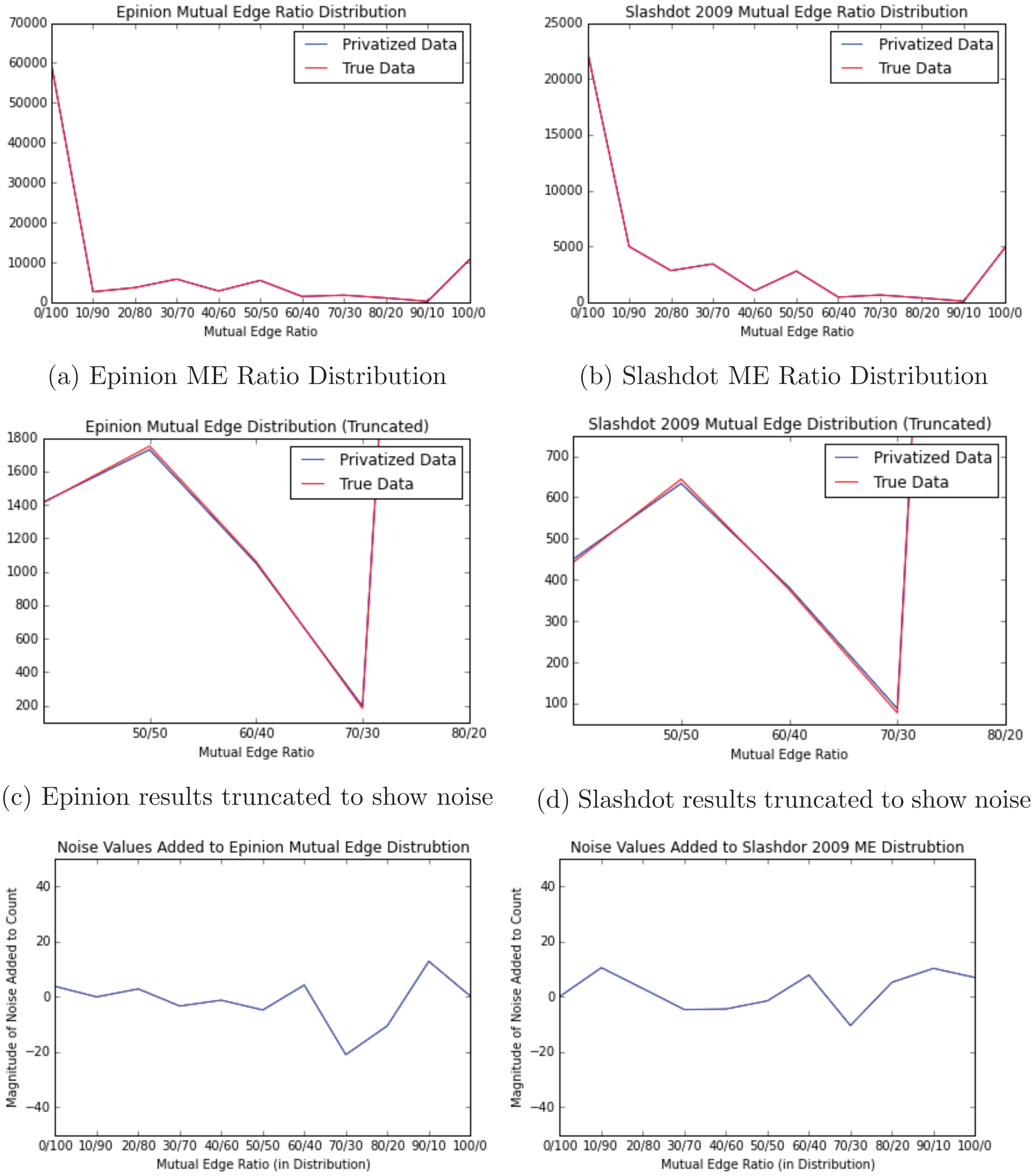
(e) Epinion privatization noise

(f) Slashdot privatization noise

Fig. 4.11.: Positive/Negative Edge Ratio (P/N) Distribution data for the Slashdot Zoo and Epinion Web of Trust networks, privatized under contributor-privacy

Privatized Results Comparisons

To perform a privacy-preserving comparative analysis of two networks, we normalize the privatized results of the first two analyses according to Definition 4.5. The



(a) Epinion ME Ratio Distribution

(b) Slashdot ME Ratio Distribution

(c) Epinion results truncated to show noise

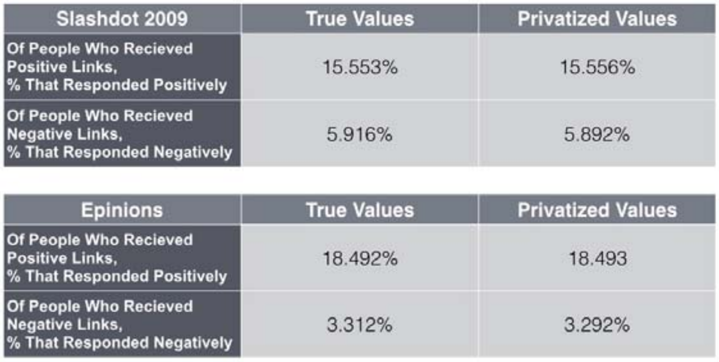
(d) Slashdot results truncated to show noise

(e) Epinion privatization noise

(f) Slashdot privatization noise

Fig. 4.12.: Mutual Edge Ratio (ME) Distribution data for the Slashdot Zoo and Epinion Web of Trust networks, privatized under contributor-privacy

resulting plots are presented in Figure 4.14 and 4.15. Results from the third analysis appear in Figure 4.4.2.



Slashdot 2009	True Values	Privatized Values
Of People Who Recieved Positive Links, % That Responded Positively	15.553%	15.556%
Of People Who Recieved Negative Links, % That Responded Negatively	5.916%	5.892%

Epinions	True Values	Privatized Values
Of People Who Recieved Positive Links, % That Responded Positively	18.492%	18.493
Of People Who Recieved Negative Links, % That Responded Negatively	3.312%	3.292%

Fig. 4.13.: Percentages of individuals who received responses In kind To a majority of their extended positive and negative edges

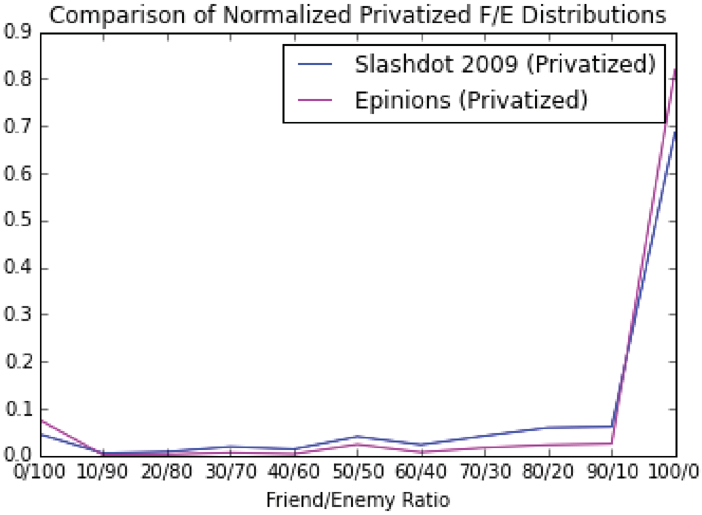


Fig. 4.14.: Comparison of normalized privatized positive/negative edge ratio distribution between Epinion and Slashdot networks.

To demonstrate that privatized social network analysis can provide useful insights into the social dynamics of signed networks, we now address the three questions we proposed above.

First, we consider whether individuals are likely to be more positive (listing more friends than enemies) or negative (listing more enemies than friends) in each network (see Figure 4.14). Interestingly, the positive/negative edge ratio distributions from

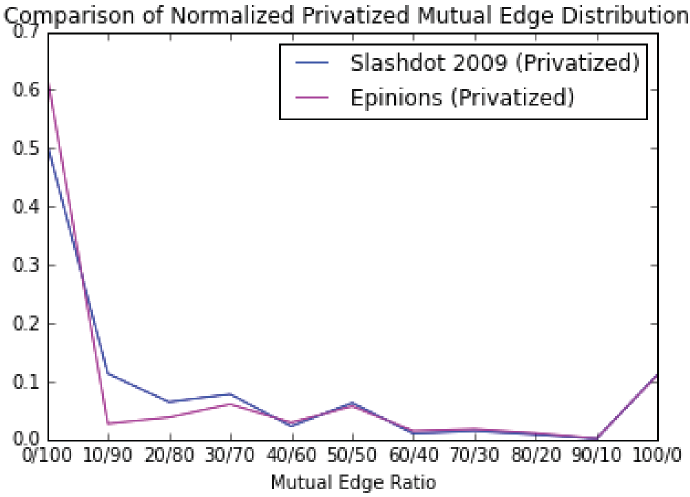


Fig. 4.15.: Comparison of normalized privatized mutual edge ratio distribution between Epinion and Slashdot networks.

both the trust and the social networks have a very similar overall structure: A few individuals list only negative relationships (distrusted reviewers and foes); a bump in the distributions at 50/50 indicates that another group of individuals precisely splits their edges between positive and negative labels; a gradual increase in the second half of the distribution indicates that many people list more friends than foes; and a final sharp rise indicates that very many people list only positive edges. The fact that these basic patterns hold in the distributions of both networks implies that even though the edges are given with different intent across different populations, lighthearted 'friend' and 'foe' status assigned in the Slashdot Zoo network and serious trust/distrust status registered in the Epinions Web of Trust, individuals have a similar overall approach to positive and negative relationships in both contexts. However, we note that the Epinions distribution is slightly more extreme, with more individuals listing only distrust or only trust, while the Slashdot network has slightly more weight in the middle of the distribution.

Our second question related to the relative reciprocity of the networks: were individuals extending an edge in the social network or in the trust network more likely to receive a mutual edge in response? In this analysis we did not consider the sign of the edge; reciprocity with respect to relationship sign will be considered in the third analysis. Again, the two distributions are strikingly similar, with two exceptions: First, we note that individuals in the Epinions network were more likely to participate in zero mutual edges (declining to rate anyone who had rated them)(see Figure 4.15). Recall that in the contributor-privacy framework, data is only collected from individuals who participate in the data-set, and thus all of the individuals in the Epinions distribution have submitted at least one trust rating for another reviewer. When an Epinions member in our data-set reciprocates 0% of her received edges, she is still actively participating in the network by extending trust relationships to others, while ignoring her own received trust relationships. Potentially this set of users is interacting with the network more as a rating system (ensuring that the reviewers they like are promoted in their feeds) than as a traditional social network (forming networks of mutual relationships with other members). We also note that the Slashdot network, by contrast, has a greater percentage of nodes which reciprocate only a small percentage, 10%-20%, of their received edges. One possibility is that this indicates a group of individuals who participate both in a few small clusters of reciprocating friendships/foeships and also extend many unreciprocated edges out beyond their friend group (possibly to a few high centrality nodes, or to random individuals of passing interest). As in the degree distribution, expanding this analysis into a two-dimensional histogram would allow us to explore this question further, without increasing sensitivity. Dimensions we might explore include the number of local (within the ego-network) communities each node participates in, or the number of high-degree nodes among each node's neighbors.

Finally, we consider reciprocity with respect to edge sign (see Figure). Here we see that people who extended positive edges were more likely to have a majority of those edges reciprocated in the Epinions network than in the Slashdot network, while those that extended negative links were more likely to have a majority of those links reciprocated in the Slashdot network than in the Epinions network. A plausible explanation for this behavior arises from the networks' distinct link semantics: a friend extending a 'foe' edge to an associate in jest on Slashdot may be likely to have it reciprocated in kind, while distrust ratings in the Epinion network may be ignored (especially if the distrusted users are false "astroturf" reviewers who are reimbursed by the sites they review and may be unlikely to participate in the trust web themselves). By contrast, mutual trust (positive) relationships may be actively encouraged by a Web of Trust network that gives trusted reviews high priority in a user's information feed.

4.4.3 Small Undirected Network: Facebook Ego-network

In our third analysis set, we look at a smaller, undirected network and a set of analyses that require greater privatization noise.

Data-sets

Our last data-set is an anonymized ego-network taken from the Facebook social network, with the ego-node itself omitted: Given a specific anonymous member of Facebook, this network was created by adding a node for each of the member's friends (but not the individual himself/herself), and then including all edges that connect the member's friends. All edges are undirected in the Facebook network because Facebook's policy enforces mutual friendships. This produces a relatively small network with 534 nodes and 9,626 edges (an average degree of 18). Contributors in this context are the individual nodes in the network (this excludes the original ego-node which was used to create the sampled graph), and the contributor data we will focus on in this analysis will be each node's knowledge of the relative popularity of their neighbors (represented by node degree).

Privatized Analysis

In this analysis we will explore a harder to privatize social network analysis scenario. To demonstrate the effect of output size on analysis privatizability we perform two independent analyses (without summing total sensitivity) over the same set of contributor data; our second analysis has output size quadratic relative to the first analysis' output size. In order to produce results with good utility, we vary our privacy parameter ϵ (we use $\epsilon = ln(2) \approx 0.69$ up to this point in Chapter 4, and throughout Chapter 5). We consider two choices for this increase: $\epsilon = (3/2)(ln2) \approx 1.04$ and $\epsilon = 3(ln2) \approx 2.08$. We note that larger values of ϵ appear in existing differentially private social network analysis literature [47], [19].

Both analyses draw on the same information from contributors: a list of each contributor's three most popular (highest degree) friends. Each analysis has a sensitivity of 3.

The first analysis compiles this information into a 'popularity distribution' which counts the number of times each node was included in a contributor's list. The privatized results are then post-processed, removing all nodes with a privatized popularity count below the threshold of 20 and forming a simple anonymous list of the popular individuals. This anonymous list by itself provides relatively little information about the underlying social network, beyond an estimate of the number of nodes with high centrality.

We can instead produce more detailed information from this data, at the cost of a large output size, by applying the Popularity Graph algorithm (which is described in detail in Section 4.2.4). The resulting, post-processed popularity graph gives additional structural information about the connectedness (Ie, the number of mutual friends) between the high centrality nodes that are submitted by contributors.

We reiterate that this analysis is not appropriate for all privacy contexts; the subject data sensitivity is quite large in this analysis ($\delta F(s) \in O(n)$, a function of the maximum number of votes that may be cast across all edges in the popularity graph that include subject s). Contributor-privacy protects the friendship information submitted by individual members of the network, but the set of high-centrality nodes (that the contributors identify communally) may be re-indentifiable in the anonymous popularity graph. As always, although privacy-preserving data-mining protects individuals' contributed data, the consequences of releasing the aggregate results is an ethical question that must also be considered. A popularity graph of a sexual relationship network would not be appropriate. One example of an appropriate use-case would be a popularity graph of a corporate office, where individual's reported relationships with their immediate coworkers require protection, but results identifying the most influential individuals are not problematic.

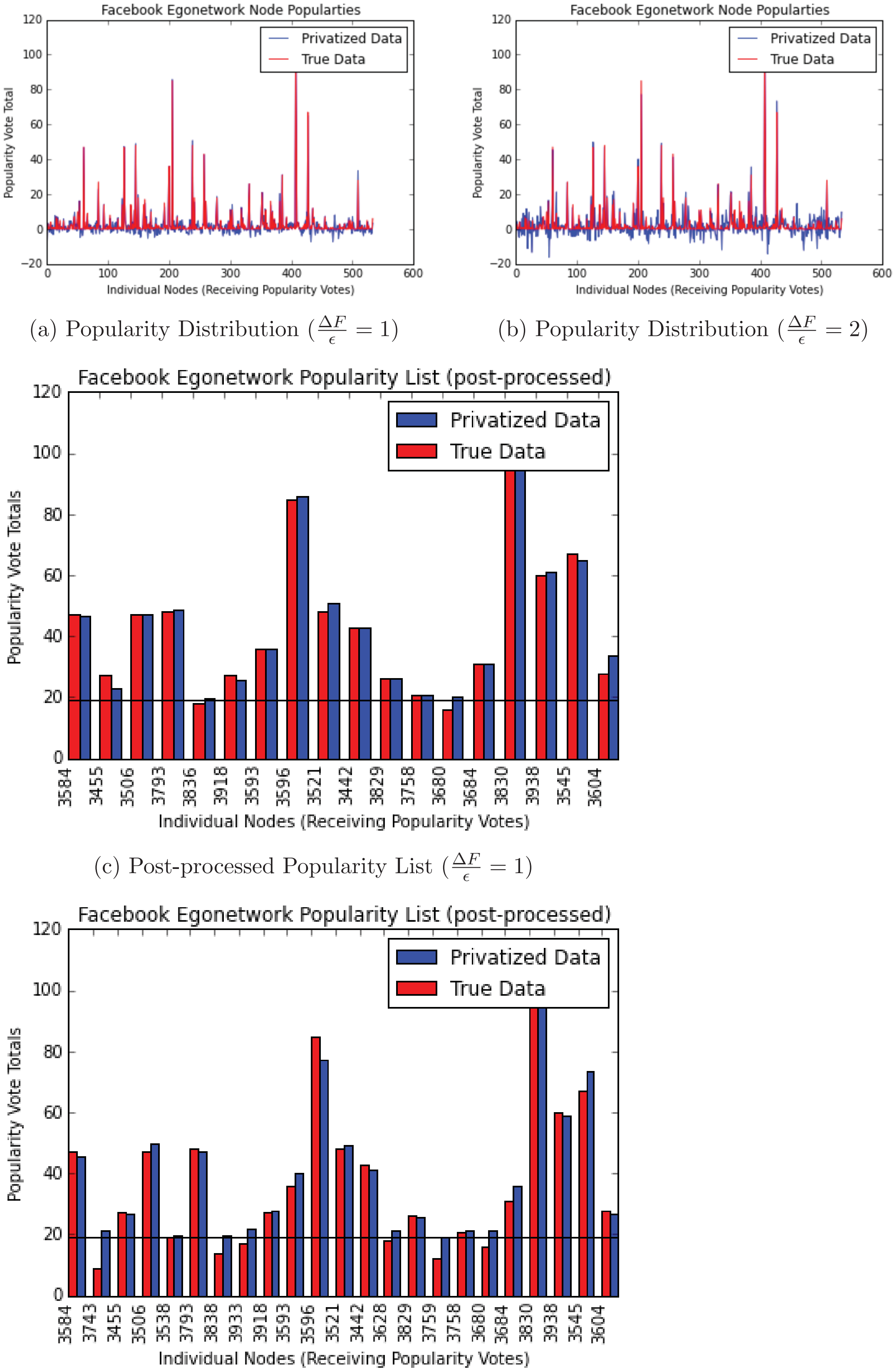
Discussion of Results

Figure 4.16 displays the results (both raw and post-processed) at both values of epsilon. The post-processing cut-off threshold is represented by the horizontal black bar at y = 20. Recall that, although the analysis set in previous section was performed with $\frac{\Delta F}{\epsilon} = 4$ which is greater that the current analysis, the visible effect of the noise on privatized results was much smaller. As we discussed in Section 4.3 (see Table 4.6), output size has a significant effect on the degree to which laplacian noise effects analysis utility; the greater output size combined with a smaller data-set (reducing the scale of the output values) is the cause of the increase in perceptible noise in these plots.

However, although the popularity distribution results are noisier than previous analysis sets, we are still able to learn about the population of interest. The post-processing eliminates most noise values, and preserves all large true data-values. At the greater epsilon value, we see that the privatized popularity list (the set of nodes whose popularity vote bars fall above the threshold line) is nearly identical to the non-privatized data; the privatized list includes two additional nodes, 3836 and 3680, that would have fallen slightly below the threshold without the addition of noise. At the lower value of epsilon, the privatized list includes seven values that would have fallen below the threshold without additional noise (and in fact two nodes, 3743 and 3759, would have fallen well below the popularity threshold without assistance); however again, all large true data values are preserved and nodes with very large popularity in the true data are easily recognizable in the privatized distribution.

Although the privatized data-sets contain some false-positives (values that would have fallen below the cut-off threshold without the addition of significant positive noise), there are no false-negatives (values that would have fallen above the threshold without the addition of significant negative noise). Our proposed social network analyses fall into two broad categories: Analyses such as LCC distributions that require small output sizes and see small impact from noise addition, and analyses such as degree-distributions that have larger output sizes but contain only a sparse set of large positive values in the non-privatized data. Because noise values are sampled independently of each other, there is a low probability that the few large negative values sampled from the noise distribution will occur at the same locations as the small, sparse set of significantly large true values. The lower probability of false-negatives in these analyses is demonstrated empirically in this analysis-set and in the analysis sets of Chapter 5.

Figures 4.17 displays the results of the popularity graph analysis (ie, the popularity graph's edge weights) at the two values of epsilon. Again, the horizontal black line indicates the post-processing cut-off threshold. Note that with the smaller value of epsilon, the increased output size requires sufficiently many noise samples such that many very large values are sampled, overwhelming the original data. However, with the higher value of epsilon, our post-processed results are very similar to the non-privatized data. The popularity graph itself is depicted in Figure 4.18 with edge color indicating edge weight. Lightweight edges represent connections between high centrality nodes which were observed by fewer individuals; these edges indicate weaker connections in the true network and are also more susceptible to privatization noise (for example, a small amount of negative noise might remove (3442,3455) from the privatized graph. However, the strong edges in the non-privatized graph are also strong in the privatized graph. The popularity graph shows three clusters of popular nodes; the second cluster is dominated by two strong edges connecting popular node 3596 with nodes 3545 and 3830. These three nodes all appear among the nodes with high weights in the popularity list of the previous analysis, however the popularity graph offers insight into their relationships with each other.



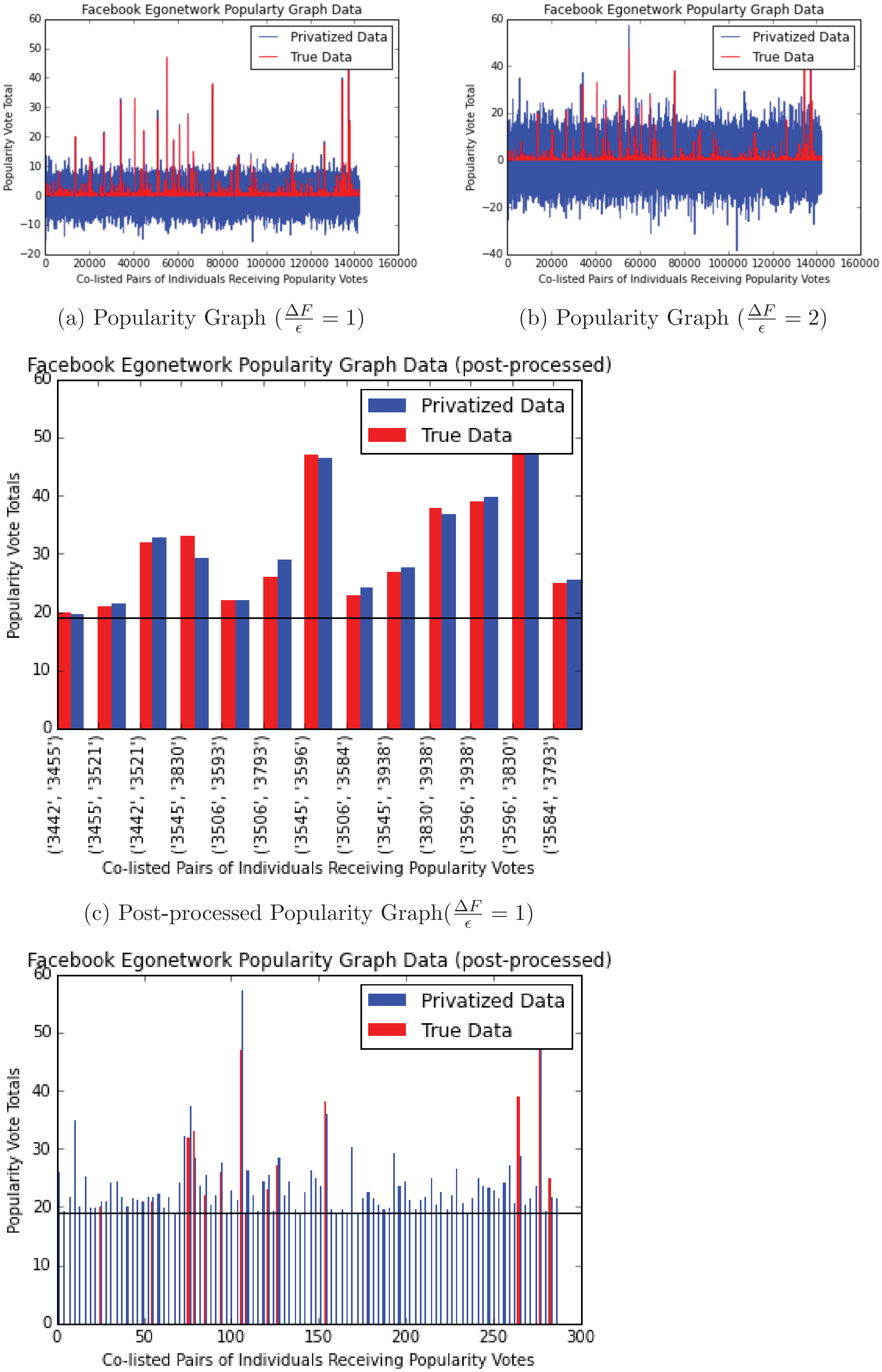
(a) Popularity Distribution ($\frac{\Delta F}{\epsilon} = 1$)

(b) Popularity Distribution ($\frac{\Delta F}{\epsilon} = 2$)

(c) Post-processed Popularity List ($\frac{\Delta F}{\epsilon} = 1$)

(d) Post-processed Popularity List ($\frac{\Delta F}{\epsilon} = 2$)

Fig. 4.16.: Popularity Distribution results for Facebook Ego-Network



(a) Popularity Graph ($\frac{\Delta F}{\epsilon} = 1$)

(b) Popularity Graph ($\frac{\Delta F}{\epsilon} = 2$)

(c) Post-processed Popularity Graph($\frac{\Delta F}{\epsilon} = 1$)

(d) Post-processed Popularity Graph($\frac{\Delta F}{\epsilon} = 2$)

Fig. 4.17.: Popularity Graph results for Facebook Ego-Network

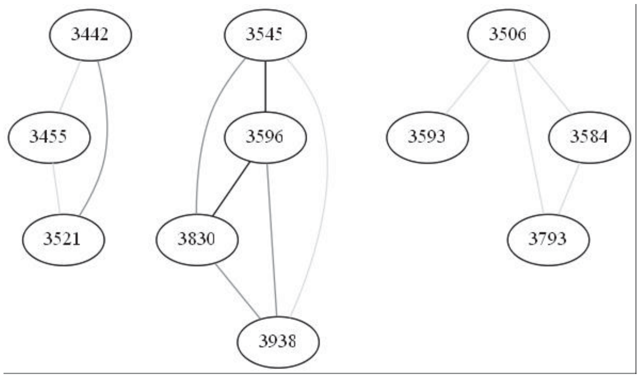


Fig. 4.18.: Privatized Popularity graph from Facebook Ego-Network: Higher edge-weights are represented by darker edge colors.

5. PARTITION PRIVACY

Many questions about social structures are naturally asked over a collection of graphs rather than one monolithic social network. Social scientists studying interpersonal interaction run experiments over large collections of small social groups, collecting social networks for each distinct group [56,57]. Collections of disjoint social networks can be implicit in larger graphs as well. Node properties such as *dormitory, major, university*, or *geographical location* can be used to partition large graphs into meaningful sets of disjoint local social networks [58]. Partition-privacy applies differential privacy to sets of graphs.

5.1 Definition

In partition-privacy, neighboring possible worlds are ones in which one subgraph is added or removed from the set of disjoint subgraphs comprising the data-set.

Definition 5.1 ***Partition-Privacy:*** *Define a partitioned graph to be comprised of separate components such that $G = \{g_i\}$ for disjoint subgraphs g_i. A privatized query Q satisfies* partition-privacy *if, for all $R \subseteq range(Q)$, and all pairs of graphs G_1, G_2 where $G_1 = G_2 - g_i$ for some $g_i \in G_1$:*

$$\frac{Pr[Q(D_1) \in R]}{Pr[Q(D_2) \in R]} \leq e^{\epsilon}$$

Partition-privacy applies when researchers wish to perform tests of hypotheses about social behavior across groups, such as “Is clustering coefficient correlated with gender in dormitory friendship structures?”. We will demonstrate in this chapter that this useful sub-class of analyses is especially amenable to privatization.

5.1.1 Privacy Analysis

Partition-privacy provides broader protection than single node-privacy: it provides protection at the level of entire social groups rather than individuals.

For functions whose sensitivity under Differential Privacy Variant 2 (see Definition 2.1) is less than or equal to their sensitivity under Differential Privacy Variant 1, partition-privacy implies k-node-privacy for nodes belonging to the same partition: Given a function F across a set of network partitions, adding or removing a set of nodes belonging to a single partition produces one of three effects: it alters the function value in at most one partition, it results in the removal of a partition from the set (if partition was comprised entirely of k nodes that were removed), or it results in the addition of a partition to the set (if k added nodes comprise a new partition). Which of these three cases produces the greater impact on the analysis results depends on the function being computed. Partition-privacy sensitivity is computed as the function value change under the addition or removal of one network partition, analogous to traditional differential privacy variant 1. In cases where this sensitivity is greater than the sensitivity computed under variant 2 (in which one partition changes its function value arbitrarily), partition-privacy will provide k-node privacy for any set of k nodes belonging to the same partition.

Recall from Section 2.1 that the sensitivity of histograms in particular is 1 under variant 1, and 2 under variant 2: When an entity is added or removed from the histogram (as in variant 1), one histogram count changes by at most 1 producing a sensitivity cost of 1. Alternatively when an entity's value is *changed* (as in variant 2), one histogram count increases by at most 1 and one histogram count decreases by at most 1, as the entity changes which count it appears in in the histogram; this produces a sensitivity cost of 2. Because the partition-private analyses in this chapter are based on histograms, they will provide k-node privacy (for nodes belonging in the same partition) with privacy-parameter $\epsilon_{node} = 2\epsilon_{partition}$.

Note that the broader protection provided by partition-privacy is important in real world scenarios in which group-level data is sensitive. For example, under degree-restricted node-privacy [19], researchers could choose to publish data that assigned an average sexual promiscuity rating to each high school in a given state, using privatized node degree data from the sexual interaction networks of students in each school. This could be seen an invasion of the students' privacy, even though no individual student's information would be distinguishable in the privatized results. With partition-privacy, each of the school networks would be protected, and only aggregate information about the distribution across the state would be publishable.

Furthermore, while existing node-private algorithms require the addition of considerable amounts of noise and provide relatively little utility in high-degree graphs (recall Section 3.3.1), partition-private analyses can require very little noise to implement. We will present a diverse selection of analyses that can be easily privatized under partition-privacy.

5.2 Basic Algorithms

In this section we will present several easy-to-use algorithmic tools that can enable social network researchers to learn about populations while guaranteeing partition-privacy. In the next section we will demonstrate the practical application of partition-private analysis on a diverse set of real world social network data-sets.

5.2.1 Triangle Count

In applications that require a collection of disjoint social networks, even more detailed privatized analysis is possible. Partition-privacy allows arbitrary analysis of disjoint subgraphs in the data-set and then privatizes the aggregation of the independent results. Assume an analysis has been performed on each disjoint subgraph, producing either a numerical result with a publicly known range (e.g., the global clustering coefficient of the graph), a category result (the gender of the dorm represented by the graph), or any combination of numerical and categorical results. The collection of graphs may now be viewed as a collection of multi-attribute data points. Removing or adding one graph from the collection is equivalent to removing or adding one of these data points; we can apply traditional differential privacy techniques to this set of independent data points as though we were working with tabular data over individuals. Two low-sensitivity techniques are very useful here: histograms and privatized means. We will demonstrate the application of these techniques in the examples below, beginning with an application of partition-privacy to triangle-count data.

The global clustering coefficient is the proportion of wedges in the graph (where one person has a two friends) that are closed to form a triangle (i.e., the two friends are also friends with each other); formally, $CC(G) = \frac{3*[number\ of\ triangles\ in\ G]}{[number\ of\ wedges\ in\ G]}$. A graph with no triangles has a clustering coefficient of 0; a clique has a clustering coefficient of 1. The clustering coefficient of a graph is a useful normalized measure of its social cohesion. However, it is difficult to draw meaningful conclusions about the population being studied using one piece of data in isolation. Given a collection of social networks, we can identify meaningful patterns of behavior by comparing clustering coefficients across networks.

Assume we want to examine how attribute X of a social group affects its degree of social cohesion. For example, we could study the relationship between the gender of a college dormitory and the clustering coefficient of the social network within the dorm. Given a data-set consisting of a collection of social networks for each possible value of X (a set of male, female and co-ed dorms), we first compute the global clustering coefficient over each individual network. We can then compute the mean of the clustering coefficients for each value of the attribute X, add noise to privatize the result, and release the privatized means (see Figure 5.1).

The mean of a set of bounded numerical values has low sensitivity when the number of values is publicly known. Consider the mean $MaleDormsClustering = M/N$ where $M = \Sigma_{G \in MaleDorms} clustering\_coefficient(G)$ and N is the number of

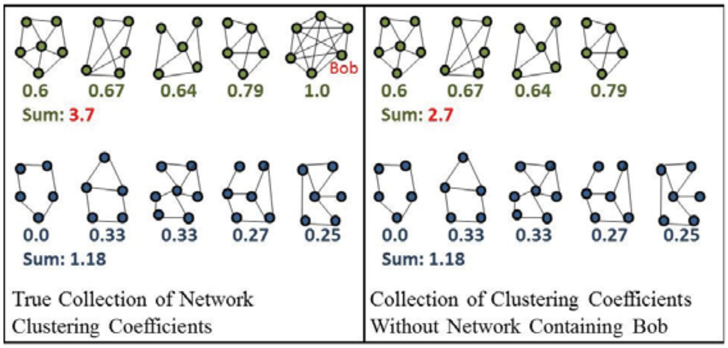
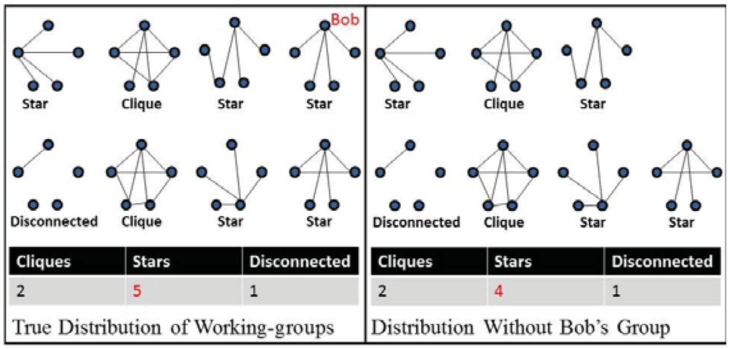

Bob
0.6
0.67
0.64
0.79
1.0
Sum: 3.7
0.0
0.33
0.33
0.27
0.25
Sum: 1.18
True Collection of Network Clustering Coefficients
0.6
0.67
0.64
0.79
Sum: 2.7
0.0
0.33
0.33
0.27
0.25
Sum: 1.18
Collection of Clustering Coefficients Without Network Containing Bob

Fig. 5.1.: Two collections of networks (blue and green) and their clustering-coefficients: Removing or altering one graph from the partitioned graph set only affects the numerator of one collection's mean by one.

male-only dorms in the data-set. If N is publicly known (for instance, because each university's dorms are listed on their website) we can safely skip adding noise to this value and focus on privatizing only the numerator M without reducing the privacy of the result [59]. Since M is a sum of clustering coefficients that have values in the bounded range [0,1], adding, removing or altering one clustering coefficient will alter the sum M by at most 1. Thus the sensitivity of the sum M is 1, and the value $\frac{M+Lap(1/\epsilon)}{N}$ will be differentially private. Note that the noise added to the true values of $MaleDormsClustering$ has a standard deviation of only $Lap(1/\epsilon)/N$.

5.2.2 Degree Distribution

Partition-privacy can also enable privatized analysis of degree distribution data. Consider the context in which a researcher performs an experiment to directly study behavior patterns in small social groups. A common technique is to assign people to small groups where they must work cooperatively to solve problems [56, 57]. Interpersonal communications in each group are monitored and analyzed. Raw communication data can be transformed into social network graphs by adding edges between nodes that communicate frequently. In small groups, different degree distributions will indicate different patterns of cooperation; for example, groups may have one



Cliques	Stars	Disconnected
2	5	1

True Distribution of Working-groups

Cliques	Stars	Disconnected
2	4	1

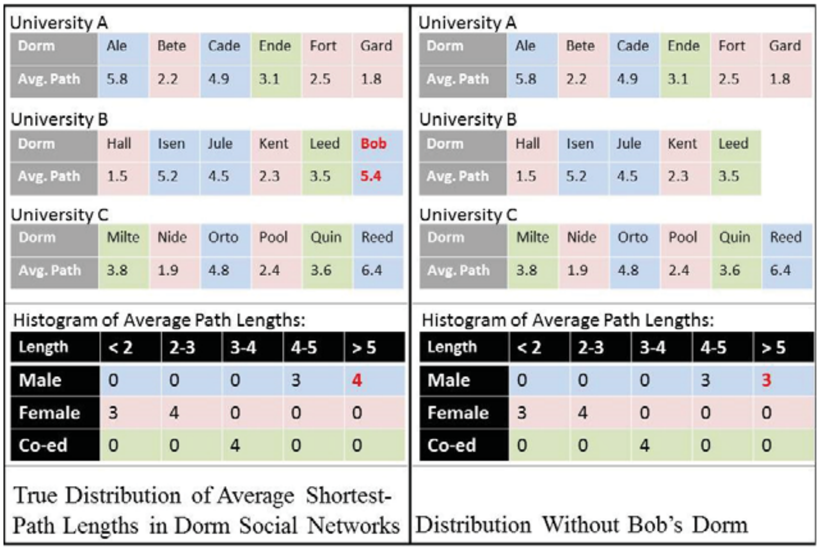
Distribution Without Bob's Group

Fig. 5.2.: Removing or adding one graph only affects the count in one histogram category by one.

high-degree 'leader' centralizing communication, or they might cooperate equitably together producing a near clique graph (see Figure 5.2). These degree-distribution categories may be affected by the group's context (e.g., working in person, or online), and they may affect the group's performance on the assigned task. When degree-distributions help us attach a meaningful category label to individual networks, we can use a privatized histogram to safely release the distribution of these labels across the set of networks. If desired, we can further partition this histogram using properties such as the group's context or performance score to create more informative multi-dimensional histograms (for an example of a multi-dimensional histogram, see Figure 5.3). As described in section 2.1, histograms have a sensitivity of only 1 and may be safely released by adding Laplacian noise calibrated to that sensitivity to each count.

5.2.3 Path-length Queries

A noteworthy property of partition-privacy is that it does not exhibit the high sensitivity to path length queries that constrains other forms of graph privacy. Although removing a bridge will drastically affect path lengths in a given network, it will only affect *one* network in the collection of small disjoint networks that comprises



University A

Dorm	Ale	Bete	Cade	Ende	Fort	Gard
Avg. Path	5.8	2.2	4.9	3.1	2.5	1.8

University B

Dorm	Hall	Isen	Jule	Kent	Leed	Bob
Avg. Path	1.5	5.2	4.5	2.3	3.5	5.4

University C

Dorm	Milte	Nide	Orto	Pool	Quin	Reed
Avg. Path	3.8	1.9	4.8	2.4	3.6	6.4

Histogram of Average Path Lengths:

Length	< 2	2-3	3-4	4-5	> 5
Male	0	0	0	3	4
Female	3	4	0	0	0
Co-ed	0	0	4	0	0

True Distribution of Average Shortest-Path Lengths in Dorm Social Networks

University A

Dorm	Ale	Bete	Cade	Ende	Fort	Gard
Avg. Path	5.8	2.2	4.9	3.1	2.5	1.8

University B

Dorm	Hall	Isen	Jule	Kent	Leed
Avg. Path	1.5	5.2	4.5	2.3	3.5

University C

Dorm	Milte	Nide	Orto	Pool	Quin	Reed
Avg. Path	3.8	1.9	4.8	2.4	3.6	6.4

Histogram of Average Path Lengths:

Length	< 2	2-3	3-4	4-5	> 5
Male	0	0	0	3	3
Female	3	4	0	0	0
Co-ed	0	0	4	0	0

Distribution Without Bob's Dorm

Fig. 5.3.: With a set of graphs, histograms can be used to release information about the relationships between multiple variables, including path lengths, with low sensitivity.

the data-set for a partition-privacy application. This enables privatized analysis for a wide variety of graph properties that are otherwise too revealing to be released.

The average shortest-path distance for a network is a measure of its connectedness. Given a collection of networks, we can find the average shortest-path length for each network and aggregate the results into a histogram, giving us information about the patterns of graph-connectedness across our data-set (see Figure 5.3). As the sensitivity of a histogram is just 1, the results can be privatized by adding a relatively small amount of noise to each count. The same technique can be used on any numerical or categorical graph property: we can privatize the distribution of maximum centrality scores, number of bridges per graph, or even graph diameters. This flexibility of application is one of the primary advantages of partition-privacy.

5.3 Utility Analysis

We note that, as in the previous chapter, the basic analysis algorithms proposed above have low sensitivity. Thus, the general utility analysis presented in Section 4.3 applies to partition-private analysis tools as well as contributor-private tools, and we direct the reader to the previous discussion. The empirical work in the subsequent section demonstrates analysis utility for a diverse set of four analyses over three moderately-sized sets of partitioned networks.

5.4 Practical Application

The partition-private analysis techniques described in this chapter have been designed to minimize analysis sensitivity and output size, such that privacy can be achieved with relatively little added noise. Additionally, partition-privacy itself provides a strong privacy guarantee, protecting entire subgraphs rather than solely nodes or edges, while simultaneously enabling privatized implementations of analyses such as community-detection and path-length metrics too sensitive to be performed under previous edge-privacy and node-privacy standards. To demonstrate the utility of partition-private techniques we now perform an in-depth analysis set on three partitioned networks.

Through this analysis set, we will explore one interesting question: What happened to Friendster?

5.4.1 Data-Sets

We will investigate this question through the three network partition sets [1], taken from the Stanford Large Network Dataset Collection [55]. These networks are available publicly online as anonymized edge and node sets and have been referenced in a wide body of social network research. Because they have been previously published in

[1] Where the original data was not a strict partition, we preprocessed the data by assigning nodes that appeared in several groups to a single, randomly selected group in that set.

simply-anonymized form, we will include both privatized and non-privatized analysis results for comparison purposes.

For these experiments we work with partition sets taken from three large networks: The DBLP online bibliography of publications in Computer Science, as well as two online social networks (OSN)–LiveJournal and Friendster. Although LiveJournal lost popularity in the United States in the mid-2000's during the rise of Facebook, it continues to be in widespread use internationally. Friendster, by contrast, was dismantled as social network in 2011 and continues today only as an online gaming website. The Friendster partition set was collected near the end of the OSN's lifespan. From each network, we consider the largest 5000 'ground-truth' communities, defined as follows:

Data-Sets: DBLP:

- **Network:** Extensive database of publications in Computer Science, including authors and venues (conferences or journals). A network is drawn from this data by including a node for each author appearing in the data-base and adding edges between individuals who have appeared together as co-authors on the same work. Edges in this network edges reflect real world collaboration efforts.
- **Groups:** Ground-truth communities in this network are defined by connected networks of authors that have published in the same venue.

Friendster:

- **Network:** Online social network launched in 2002 which gathered over 8 million users before being abandoned and finally dismantled in 2011. This network data was collected shortly before the OSN was shut down. Nodes indicate members, and edges between them indicate OSN friendships.
- **Groups:** Ground-truth communities in this network are user-defined groups: Friendster allowed members to create groups which other members could join. The semantics of these groups is diverse: they might reflect broadly shared interests or hobbies, or particular groups of friends.

LiveJournal:

- **Network:** Online social network launched in 1999; current usage includes approximately 1.8 million active users and 39.6 million total accounts. Nodes indicate members, and edges between them indicate OSN friendships.

- **Groups:** Ground-truth communities in this network are user-defined groups: LiveJournal allowed members to create groups which other members could join. The semantics of these groups is diverse: they might reflect broadly shared interests or hobbies, or particular groups of friends.

5.4.2 Privatized Analysis

We performed the following four analyses across each of the three networks. Each analysis incurs a sensitivity cost of 1, producing a total sensitivity of 4; laplacian noise sufficient to obfuscate this sensitivity was added to every output value. The parameter $\epsilon = ln(2)$ was used throughout this analysis. Distribution cut-offs are chosen independently of the data-set to ensure no additional sensitivity cost.

Analyses

- **Average Shortest Path Distribution**: We computed the average shortest path (the mean of the set of distances computed between every pair of nodes in the group) for each group in the network, and aggregated this data into a distribution with a cut-off of 8. Recall from Chapter 3 that path-length information is not generally privatizable under edge-privacy or node-privacy standards; however, partition-privacy offers a high-privacy and low-sensitivity tool for studying these network properties. Smaller average shortest path values indicate groups in which nodes are more tightly interconnected. Results for all three networks are presented in Figure 5.4.

- **Average Local Clustering Coefficient Distribution**: We computed the average local clustering coefficient (the mean of the set of LCC's taken across all of the nodes in the group) for each group in the network, and aggregated this data into a distribution with a precision of 0.1. Larger average local clustering coefficient values indicate groups with high transitivity: groups in which any two friends of an individual are likely to also be friends with each other. These groups tend to be more socially cohesive. Results for all three networks are presented in Figure 5.5.

- **Edge Density Distibution**: We computed the edge density (the total number of edges in the group divided by the number of nodes; also known as the 'average degree') for each group in the network, and aggregated this data into a distribution with a cut-off of 30. Large edge density values indicate groups in which nodes extend a greater number of edges to other individuals. Results for all three networks are presented in Figure 5.6.

- **Community Count Distribution**: Using an implementation of the Louvain Community Detection method [60] (a popular modularity-based community detection method) provided in the community.py python library, we computed the number of independent (partitioned) community substructures in each group. We then aggregated this data into a distribution with a cut-off of 150. Community-detection has not been previously achieved with node-privacy in existing work, but partition-privacy provides a tool which allows us to study this network property with very low sensitivity and a privacy guarantee that is stronger than node-privacy. As with the schism visible in the Karate Graph, well-defined sub-communities existing within a larger group indicate the group is less unified. Results for all three networks are presented in Figure 5.7.

In general, we can see that noise had relatively little visible effect in the privatized output for this analysis set; this is a result of the low sensitivity, constrained output size, and the size of the data-sets (the number of partition groups from each network). One exception is the community count distributions for the Friendster and, to a lesser extent, the LiveJournal partition set (see Figure 5.7). Note that the scale of the yaxis, indicating the size of the plotted data, is reduced for these distribution: both partition sets had a portion of groups with large and diverse community counts, extending the tail of their distributions beyond the cut-off, and allowing the effect of the additive noise to be more significant relative to the scale of the output counts. One step to improve the level of detail available on the tails of these distributions would be to increase 'bucket-widths': reduce the granularity of the x-axis into ranges of size 10 or 20. By increasing the size of the counts that fall into each range, this would reduce the impact of noise addition (which is relative to the size of the data). However, we will retain the original axes for comparison purposes in this analysis set. Note that, although the Friendster distribution plot displays more noise, because the noise is added independently to each output the underlying curve of the distribution remains visible.

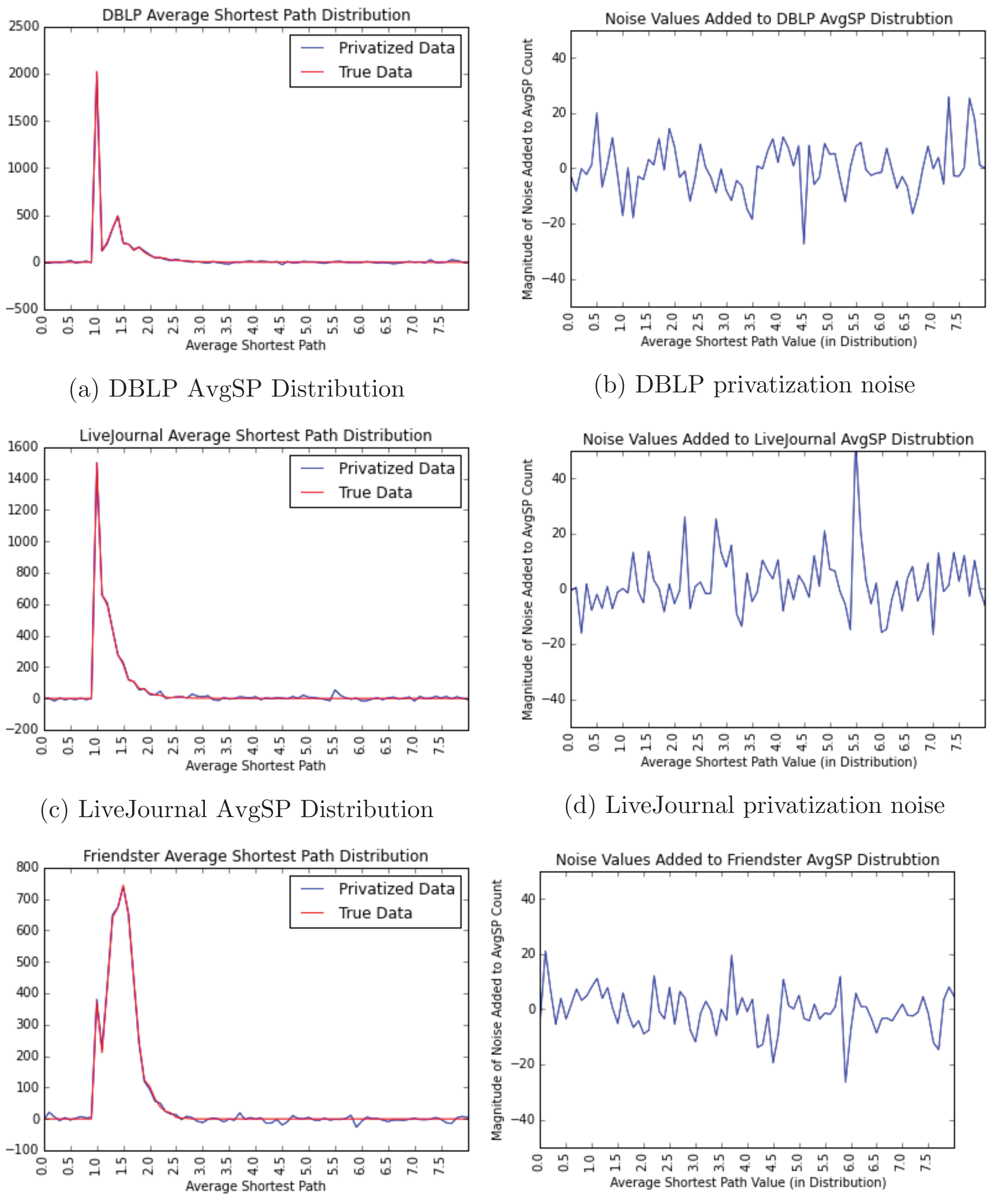
5.4.3 Conclusions

We now return to address our original question: In what ways does the Friendster group partition set differ from the more successful and longer lived networks?

We use privatized normalized distributions for comparison between networks, using the approach given in Definition 4.5 in the previous chapter.

Throughout these analysis results we see that the distributions from the longer lived LiveJournal network bear a greater similarity to the real-world collaboration DBLP network than the failed OSN Friendster.

In terms of average shortest path (see Figure 5.8), we see that both DBLP and LiveJournal distributions have considerable weight on very small AvgSP lengths (1-1.2), while the Friendster distribution has its greatest weight at a longer path length (1.5-1.6). This indicates that a majority of the Friendster groups were less tightly connected than groups in the DBLP and LiveJournal networks. It's interesting to note that the DBLP distribution also had some weight at 1.5, in a bimodal distribution:



(a) DBLP AvgSP Distribution

(b) DBLP privatization noise

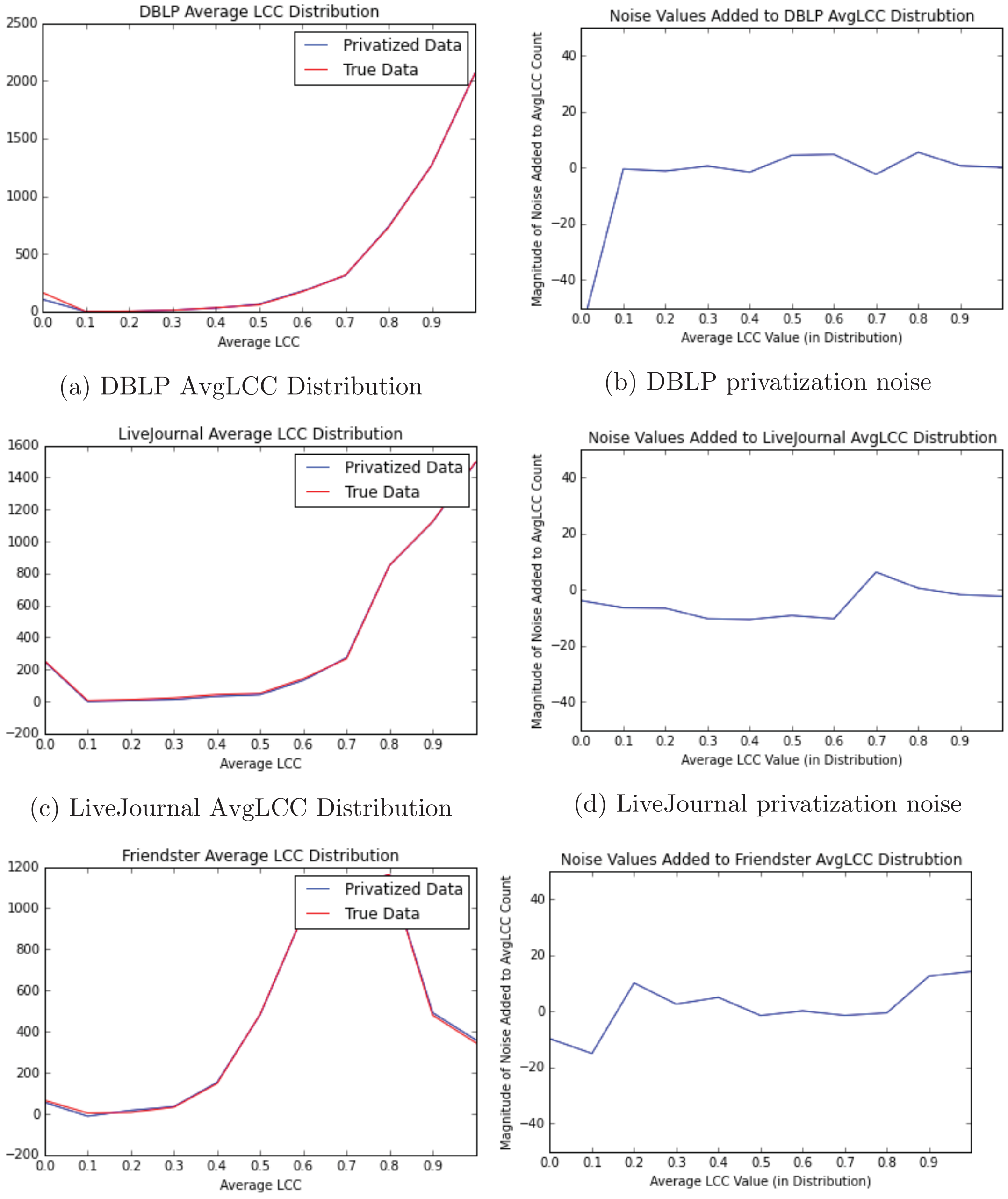
(c) LiveJournal AvgSP Distribution

(d) LiveJournal privatization noise

(e) Friendster AvgSP Distribution

(f) Friendster privatization noise

Fig. 5.4.: Results of the partition-private average shortest path distributions for the DBLP, LiveJournal and Friendster group networks



(a) DBLP AvgLCC Distribution

(b) DBLP privatization noise

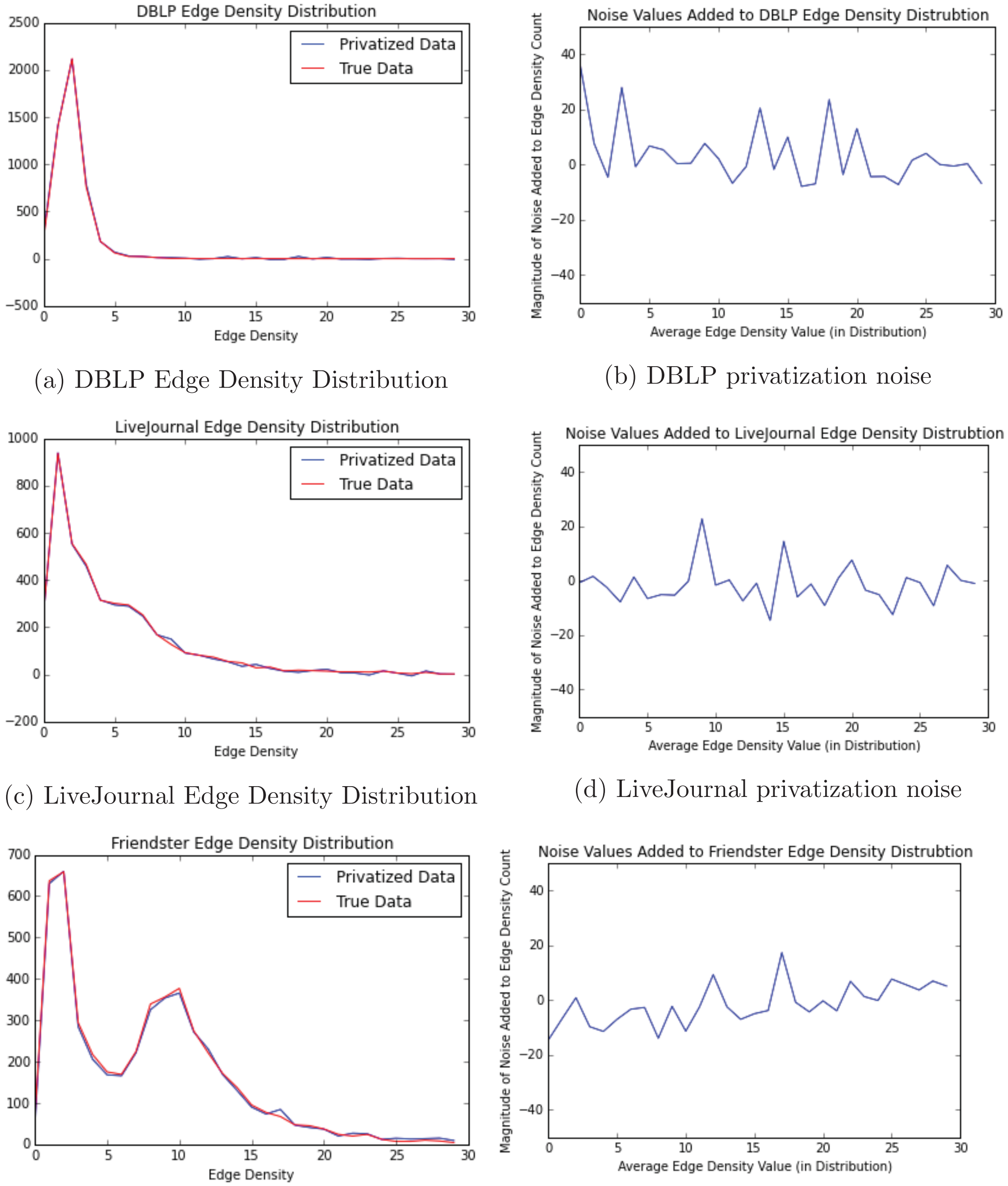
(c) LiveJournal AvgLCC Distribution

(d) LiveJournal privatization noise

(e) Friendster AvgLCC Distribution

(f) Friendster privatization noise

Fig. 5.5.: Results of the partition-private average local clustering coefficient distributions for the DBLP, LiveJournal and Friendster group networks



(a) DBLP Edge Density Distribution

(b) DBLP privatization noise

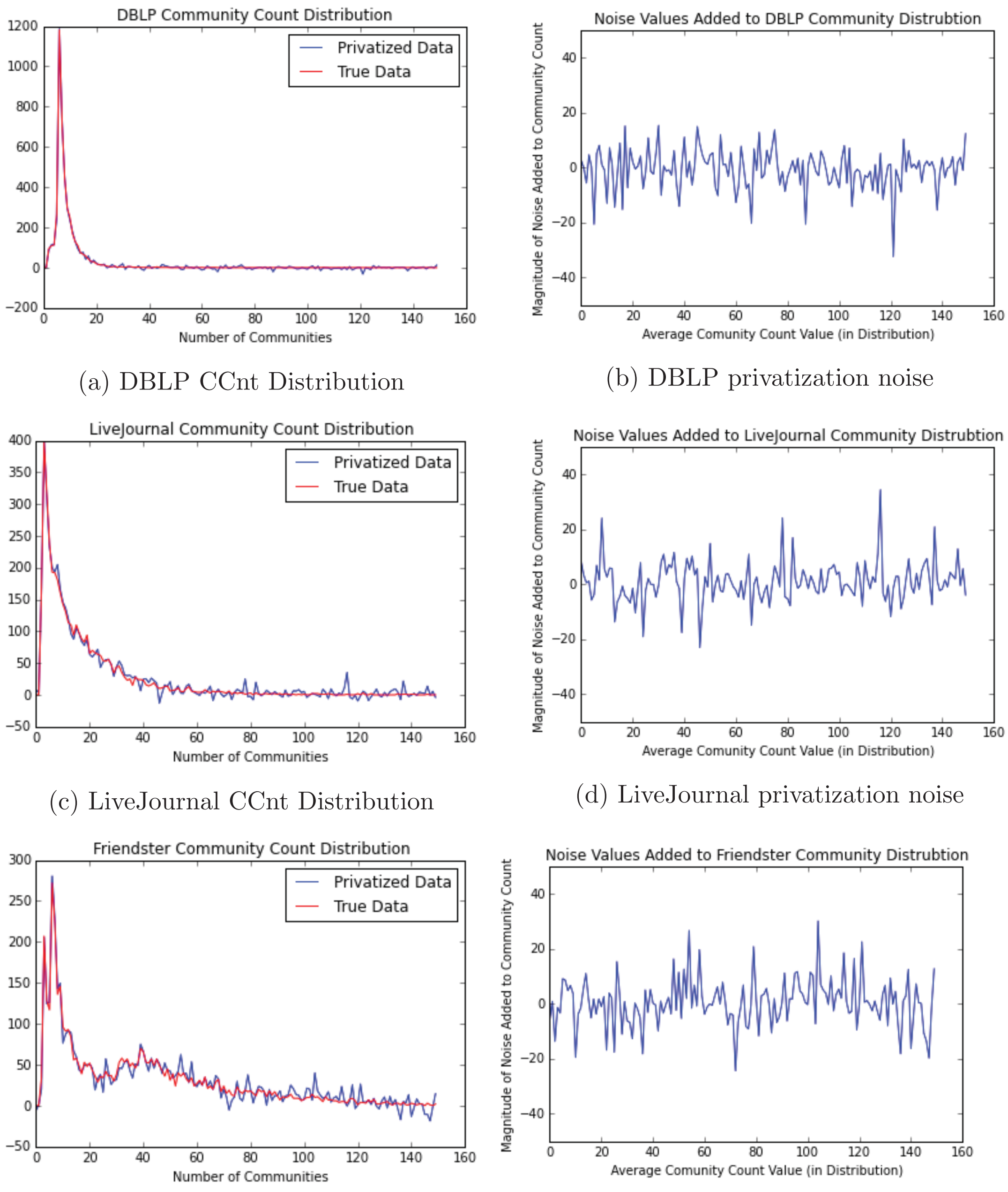
(c) LiveJournal Edge Density Distribution

(d) LiveJournal privatization noise

(e) Friendster Edge Density Distribution

(f) Friendster privatization noise

Fig. 5.6.: Results of the partition-private edge density distributions for the DBLP, LiveJournal and Friendster group networks



(a) DBLP CCnt Distribution

(b) DBLP privatization noise

(c) LiveJournal CCnt Distribution

(d) LiveJournal privatization noise

(e) Friendster CCnt Distribution

(f) Friendster privatization noise

Fig. 5.7.: Results of the partition-private community count distributions for the DBLP, LiveJournal and Friendster group networks

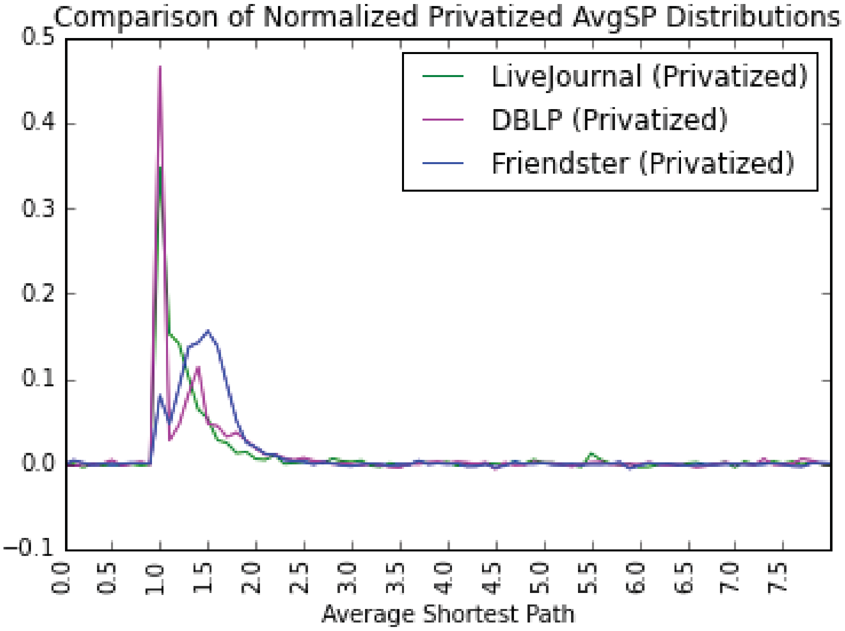


Fig. 5.8.: Comparison of normalized privatized average shortest path distributions.

this might represent the case in which a small number of distinct collaboration groups at one venue are connected by one or two bridge nodes who work with both.

In terms of average LCC (see Figure 5.9), we see that both the DBLP and LiveJournal network distributions have most of their weight at high LCC values (0.8-1.0), while the majority of groups in the Friendster network showed less transitivity, with average LCC values in the 0.5-0.8 range. This implies that both the groups reflecting the real world DBLP relationships and the groups reflecting online relationships in LiveJournal tended to be more socially cohesive than the groups in Friendster.

In terms of edge-density (see Figure 5.10), we see that many groups in both LiveJournal and DBLP had a less dense edge-set with peaks falling in the 1-5 average degree range (and a longer tail on the LiveJournal distribution). By contrast, the Friendster network showed a bimodal distribution with considerable weight at a higher edge density of 9-10. This is interesting in light of the fact that, although groups in Friendster tended to have a greater edge-density, the previous distribution

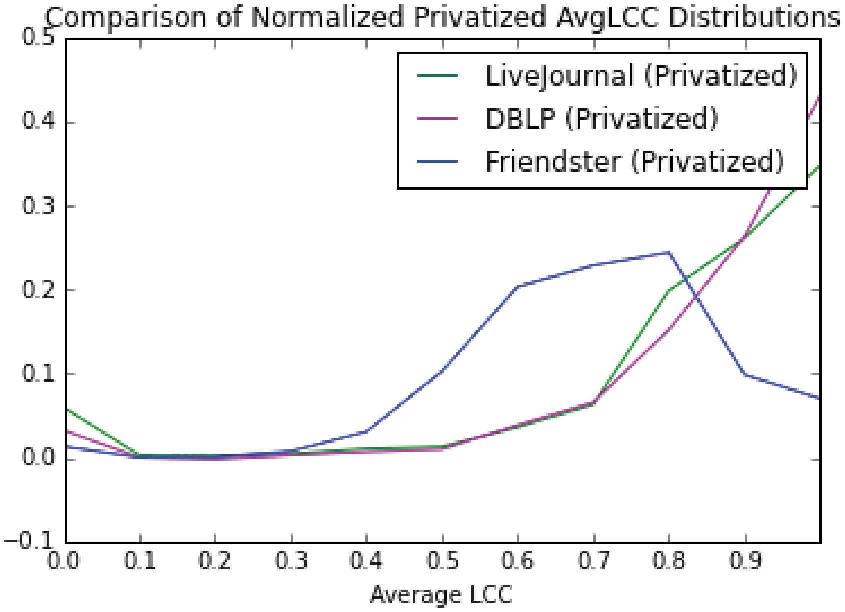


Fig. 5.9.: Comparison of normalized privatized average local clustering coefficient distributions.

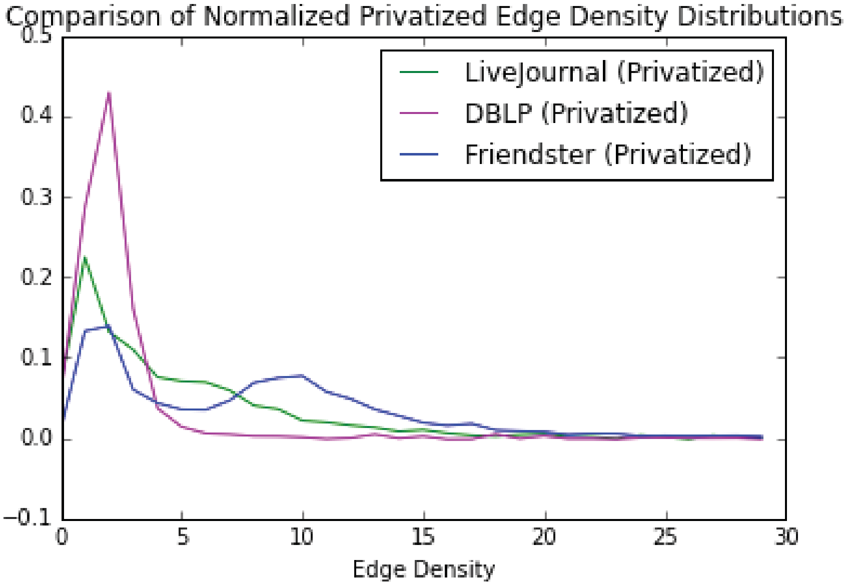


Fig. 5.10.: Comparison of normalized privatized edge density distributions.

indicates that they had less transitivity: these added edges weren't necessarily forming cohesive friendship groups. Similarly, the average shortest path results indicate that the additional edges in Friendster were not increasing the small-world property of the network by bringing outlier nodes or poorly connected subgroups into better connection with the network (and thus reducing path length).

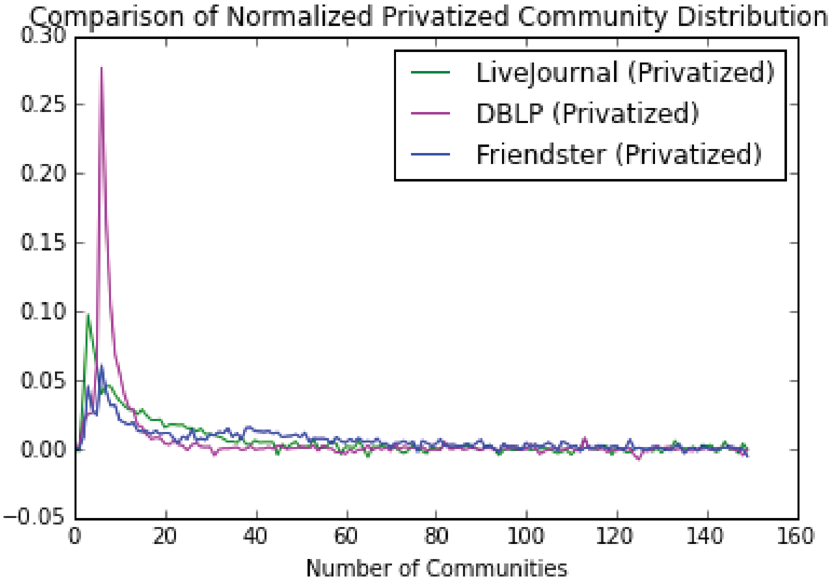


Fig. 5.11.: Comparison of normalized privatized community count distributions.

In terms of community count (see Figure 5.11), we see similar peaks in all three distributions, focused on the 5-15 range, with different weights on those peaks vrs. the tails of the distributions. The vast majority of the groups in the DBLP network had fewer than ten subgroups, while the Friendster groups were much more likely to have many (more than 20) distinct sub-communities. LiveJournal fell in between these two distributions, with more weight falling in the 1-30 range. This indicates less unity in the OSN's, with groups in Friendster being especially divided (again, despite having a greater edge density).

Overall, although this analysis set does not necessarily support causal inferences, we can hypothesize from this evidence that by the time of its demise Friendster had begun to lack many of the properties that are inherent in real world social networks: groups in Friendster had a glut of edges that did not form transitive communities, did not tightly connect nodes within the group (reducing path length), and did not unify groups (reducing the sub-community count). These edges would have had a significant presence in their members' profiles without necessarily offering the same benefits as real world relationships, acting more like random edges than social ties into well-defined communities. If time-series data were available, it might be very informative to see how these properties evolved as the Friendster OSN declined.

6. STATISTICAL SIGNIFICANCE

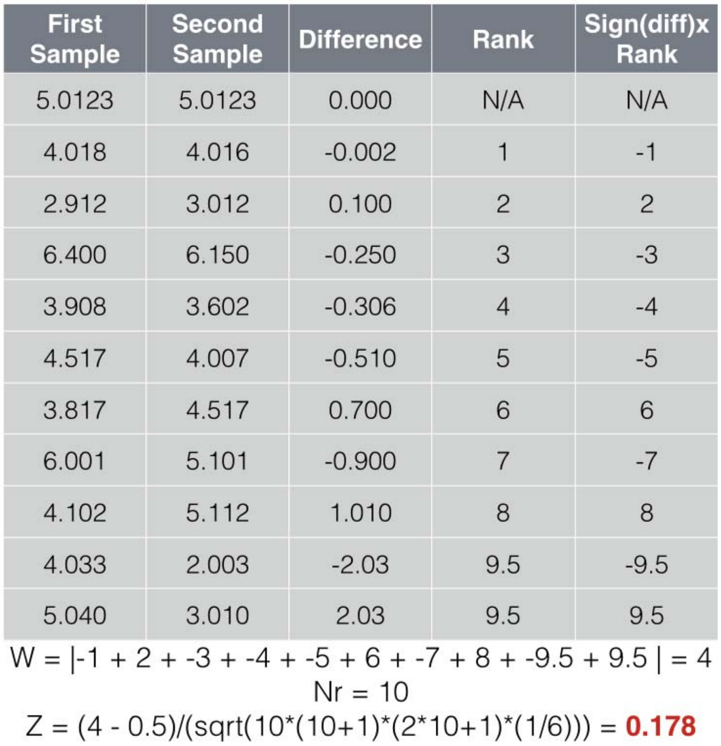
6.1 Introduction

The ability to perform statistical significance testing is vital in real world social science applications. In order to draw reliable inferences from data analysis results it is necessary to distinguish between when an observed difference in two sampled data distributions is the result of a fundamental difference in the two underlying random variables being sampled, and when it is more likely the result of random sampling error (or, in our case, added privatization noise).

We now consider one real world application of social network analysis–intervention in organizational networks. An organization, such as a large corporation or university, may collect data on the relationships between their members, and use research on the correlation between network properties and individal/organizational success to determine the overall 'health' of their network. For example, certain network properties have been shown to be significant in predicting women's ability to resist the negative effects of sexism in corporate environments [61], [62]. The organization's leadership may then introduce programs to address any observed concerns, such as activities intended to foster tighter relationships or more diverse relationships among organization members. And finally, a second network is collected over the same set of individuals in order to determine the effectiveness of the intervention program. This produces two sets of sampled data over the same set of individuals, a 'before' network and an 'after' network, or a 'paired-sample' data-set.

In this chapter we present a differentially privatized approach to determining statistical significance on paired-sample data, using the Wilcoxon Signed-Rank test [63]. In addition to being applicable to the paired-sample data generated by the network interventions described above, this test is well-suited to social network analysis because, unlike the widely-used Student's T-test, the Wilcoxon Signed-Rank Test does not require the underlying data to have a normal distribution. As can be observed in the empirical results of the preceding two chapters, social network data does not in general produce a normal distribution.

6.2 Background: Wilcoxon Signed-Rank Test



First Sample	Second Sample	Difference	Rank	Sign(diff)x Rank
5.0123	5.0123	0.000	N/A	N/A
4.018	4.016	-0.002	1	-1
2.912	3.012	0.100	2	2
6.400	6.150	-0.250	3	-3
3.908	3.602	-0.306	4	-4
4.517	4.007	-0.510	5	-5
3.817	4.517	0.700	6	6
6.001	5.101	-0.900	7	-7
4.102	5.112	1.010	8	8
4.033	2.003	-2.03	9.5	-9.5
5.040	3.010	2.03	9.5	9.5

W = |-1 + 2 + -3 + -4 + -5 + 6 + -7 + 8 + -9.5 + 9.5 | = 4

Nr = 10

Z = (4 - 0.5)/(sqrt(10*(10+1)*(2*10+1)*(1/6))) = **0.178**

Fig. 6.1.: Demonstration of the non-privatized Wilcoxon Signed Rank Test procedure

The Wilcoxon Signed-Rank Test was proposed by Frank Wilcoxon in 1945 [64]. It takes as input a set of paired samples, generated by the same set of individuals measured before and after the administration of a 'treatment', and produces a statistic that can be used to determine whether the distribution after the treatment is significantly distinct from the distribution before treatment. The test assumes that the individuals included in the statistic are selected independently of each other, and that the measure being studied is continuous rather than discrete.

6.2.1 Test Procedure

Given a set of N individuals $i \in I$ and paired data samples (p_{i1}, p_{i2}) for each individual i, the Wilcoxon Signed-Rank Test statistic is computed as follows [63] (see Figure 6.1):

Wilcoxon Signed-Rank Test Procedure

[1] The difference $d_i = (p_{i2} - p_{i1})$ is computed for each individual $i \in I$

[2] All i with $d_i = 0$ are removed. The remaining non-zero differences comprise a *difference-set* D. We use the notation $N_R = |D|$ to refer to the 'reduced N' size of the non-zero difference-set.

[3] The difference-set is sorted into increasing order by absolute value:
$[d_1, d_2, d_3 ... d_{N_R}]$, such that $i < j \Rightarrow |d_i| \leq |d_j|$

[4] Ranks are assigned to values in the list of differences in increasing order such that $r_i = rank(d_i) = i$ (with one exception: If there exists a tie between two differences, such that $d_i = d_{i+1}$, both values are assigned the rank $(i + i + 1)/2$; In general, in a tie of size k, $d_i = d_{1+1} = ...d_{i+k-1}$ all elements are assigned their average rank $\frac{1}{k}\Sigma_i^{k-1} j$.)

[5] The absolute value of the signed rank sum $W = |W_{raw}| = |\Sigma_1^{N_R} sign(d_i) * r_i|$ is computed

[6] The denominator $\sigma(N_R) = \frac{1}{\sqrt{\frac{N_R(N_R+1)(2N_R+1)}{6}}}$ is computed

[7] The test statistic $Z = \frac{W-0.5}{\sigma(N_R)}$ is computed

[8] If $N_R \geq 10$ the value Z is compared against a normal Z_{crit} table, otherwise the value is compared against a table explicitly computed for ranked-sums. [63]

6.3 Privatization

6.3.1 Computing Sensitivity

We now compute the function sensitivity of the Wilcoxon Signed-Rank test under contributor-privacy, using a second-variant definition of neighboring worlds: ie, the total number N_R of individuals in the difference set is static, but the value d_i of a single individual may vary arbitrarily among non-zero values (see Definition 2.1).

Theorem 6.1 *The sensitivity of the statistic Z_{N_R} produced by the Wilcoxon Signed-Rank test with fixed difference-set size N_R is $2N_R/\sigma(N_R) = \frac{2N_R}{\sqrt{\frac{N_R(N_R+1)(2N_R+1)}{6}}}$*

Proof: *We alter the value of individual j as follows: $(r_j = v_1) \rightarrow (r_{j'} = v_2)$, for $v_1, v_2 \neq 0$. When the value of j is altered such that its signed rank r_j changes from v_1 to v_2, the value of the raw rank sum (before the absolute value) changes by $(v_2 - v_1)$. Because the number of non-zero differences (N_R) has not been altered, the value of the denominator $\sigma(N_R)$ remains unchanged. WLOG, the change in the numerator is maximized for $v_1 = -N_R$, $v_2 = N_R$ (j is changed from the greatest positive difference value to the greatest negative difference value).*

Note that:

$\frac{W-0.5+2N_R}{\sigma(N_R)} = \frac{W-0.5}{\sigma(N_R)} + \frac{2N_R}{\sigma(N_R)} = Z_{N_R} + \frac{2N_R}{\sigma(N_R)}$

Thus the total change to the statistic Z_{N_R} is $\frac{2N_R}{\sigma(N_R)}$.

Sensitivity:$\Delta Z_{N_R} = 2N_R/\sigma(N_R)$

Given this sensitivity as a function of N_R, we note the following useful fact:

Corollary 6.2 *Laplacian privatization noise generated with parameter $\frac{2N_{R1}}{\sigma(N_{R1})\epsilon}$, which is sufficient to privatize a Wilcoxon Signed-Rank Test(WSRT) statistic $Z_{N_{R1}}$ with difference-set size N_{R1}, will also be sufficient to privatize any WSRT statistic $Z_{N_{R2}}$ with a larger difference-set size $N_{R2} > N_{R1}$.*

Proof: *We'll show that $N_{R2} > N_{R1}$ implies that $\Delta Z_{N_{R2}} \leq \Delta Z_{N_{R1}}$, and thus noise added to privatize $Z_{N_{R1}}$ will be more than enough to privatize $Z_{N_{R2}}$:*

$$\frac{2N_R}{\sigma(N_R)} \geq \frac{2(N_R+k)}{\sigma(N_R+k)}$$

$$\frac{2N_R\sigma(N_R+k)}{\sigma(N_R)\sigma(N_R+k)} \geq \frac{2(N_R+k)\sigma(N_R)}{\sigma(N_R+k)\sigma(N_R)}$$

$$2N_R\sigma(N_R+k) \geq 2(N_R+k)\sigma(N_R)$$

$$2N_R\sqrt{\frac{(N_R+k)(N_R+k+1)(2N_R+2k+1)}{6}} \geq 2(N_R+k)\sqrt{\frac{(N_R)(N_R+1)(2N_R+1)}{6}}$$

$$\frac{\sqrt{4N_R^2(N_R+k)(N_R+k+1)(2N_R+2k+1)}}{\sqrt{6}} \geq \frac{\sqrt{4(N_R+k)^2(N_R)(N_R+1)(2N_R+1)}}{\sqrt{6}}$$

$$\frac{\sqrt{4N_R^2(N_R+k)(N_R+k+1)(2N_R+2k+1)}}{1} \geq \frac{\sqrt{4(N_R+k)^2N_R(N_R+1)(2N_R+1)}}{1}$$

$$4N_R^2(N_R+k)(N_R+k+1)(2N_R+2k+1) \geq 4(N_R+k)^2(N_R)(N_R+1)(2N_R+1)$$

$$N_R(N_R+k+1)(2N_R+2k+1) \geq (N_R+k)(N_R+1)(2N_R+1)$$

$$(N_R^2+N_R(k+1))(2N_R+2k+1) \geq (N_R^2+(k+1)N_R+k)(2N_R+1)$$

$$2N_R^3+2(k+1)N_R^2+(2k+1)N_R^2+(2k+1)(k+1)N_R \geq 2N_R^3+2(k+1)N_R^2+2kN_R+ N_R^2+(k+1)N_R+k$$

$$2(k+1)N_R^2+(2k+1)N_R^2+(2k+1)(k+1)N_R \geq 2(k+1)N_R^2+N_R^2+2kN_R+(k+1)N_R+k$$

$$(4k+3)N_R^2+(2k+1)(k+1)N_R \geq (2k+3)N_R^2+(3k+1)N_R+k$$

$$2kN_R^2+(2k^2+3k+1)N_R \geq (3k+1)N_R+k$$

$$2kN_R^2+2k^2N_R \geq k$$

$$2kN_R^2+2k^2N_R-k \geq 0$$

$$2N_R^2+2kN_R-1 \geq 0$$

$$2N_R^2+2kN_R \geq 1$$

Which clearly holds for $k, N_R \geq 1$

We now address the fact that this sensitivity is a function of N_R, the number of people who were affected by the treatment in *some* fashion (either large or small, positive or negative) by the treatment. The value N_R itself may be sensitive information that cannot be published or used as a parameter in privatized analysis. The Wilcoxon Signed-Rank Test assumes data sets with continuous values in which very small difference values may be common, but zeros are relatively rare (similar to the hypothetical data in Figure 6.1). Social network statistics such as local clustering coefficients, or an individual's average edge strength over homophilous links [1] (e.g., characterized in terms of amount of email exchanged, or number of meetings), produce this type of continuous data-set. In these continuous data-sets, small levels of variation are common and the fact that $d_i > 0$ may not constitute particularly privacy-invasive information about i: this fact does not reveal either the size or the sign of d_i, leaving the possibility that d_i is very small and due to random fluctuations rather than any specific reaction to the treatment. We provide a *High-Utility* variant of our Privatized Wilcoxon Signed-Rank test that assumes limited information about N_R can safely be made public.

However, in high privacy-risk contexts N_R may be a sensitive value that requires protection. For example, a set of average edge weights in a sexual-interaction network, taken before and after a sexual education class, would require careful privacy protection. We also provide a *High-Privacy* variant of our privatized Wilcoxon Signed-Rank Test that controls disclosure about N_R. Importantly, although they may produce false negatives on smaller data-sets or when privatizing lower significance values, neither technique introduces new (unaccounted for) false positives into the privatized significance testing results.

Two Variants of the Privatized Wilcoxon Signed-Rank Test

- **High Utility:** This privatization procedure assumes that data-owners are comfortable publishing this statement: **"Our data-set contains at least $N > 30$ individuals, and if more than 30% of our data-set shows precisely zero reaction to the treatment, we will assume the treatment effect was insignificant and will not publish our results."** The privatized analysis assumes $N_R \geq 0.3N$, computes the non-privatized statistic Z, and adds laplacian noise accordingly (see Theorem 6.1) to satisfy differential privacy.

- **High Privacy:** This privatization procedure assumes that data-owners are not comfortable publishing any information regarding N_R. The privatization

[1] The strength of a woman's network connections to other women has been shown to predict career success for women in corporate organizations. [61]

scheme obfuscates the true value of N_R by first **'priming'** the difference set with $2k$ synthetic values, to ensure a minimum value of N_R. Laplacian noise sufficient to obfuscate a statistic with $N_R \geq 2k$ is then added (see Theorem 6.1), satisfying differential privacy. This approach is explained in detail in the next section.

6.3.2 Priming D for High-Risk Applications

In cases where privacy risk is high, we provide a method for ensuring a known minimum value of N_R without revealing any information about the true original value of N_R. Intuitively, this requires 'priming' the difference set by a inserting a small amount of synthetic data into the difference-set before beginning the Wilcoxon Signed Rank Test procedure. Given a data-set with difference-set D of size N_R, we take the following steps to prime D with $2k$ initial values:

Priming Procedure

[**1**] Because D is finite, there exists a number $\top$ such that $\top > d_i, \forall i$. (The actual value of this number is not significant, so no privacy leaks are induced by this observation).

[**2**] Create a new difference-set D' by adding to D: k values of size $\top$ and k values of size $-\top$.

[**3**] Compute the statistic Z' over the primed D'. Note that $N_{R'} > 2k + N_R > 2k$

[**4**] It is now possible, by Corollary 6.2, to privatize Z' by adding laplacian noise sufficient for sensitivity $2(2k)/\sigma(2k)$ without needing to know the true value of N_R.

We now consider the effect priming has on the value of the test statistic Z.

Lemma 6.3 *Priming does not alter the value of the signed rank-sum W*

Proof: *Note that the priming $-\top, \top$ values, which form a tie of size 2k, will all be assigned the same rank: $r_\top = (N_R + 2k)/2k$. Because they are at the top of the ranking, their inclusion will not alter the rankings for any of the differences in the original difference-set* [2]*. And because the two groups of k priming values have opposing signs, they cancel each other out in the sum: $kr_\top + -kr_\top = 0$, producing no effect on the signed rank-sum W.*

Given this result, it is trivial to show that priming does not introduce false positives in statistical significance tests:

Theorem 6.4 *Priming does not increase the value of test statistic Z, and thus does not introduce false positives (which appear significant when the original statistic was not large enough to be significant):*

Proof: *Using Lemma 6.3, we see that priming does not alter the numerator ($W - 0.5$) of Z. Priming does however, increase the value of the denominator, as $\sigma(N_R + 2k) > \sigma(N_R)$. This decreases the value of the test statistic. Test statistics which show significance with priming will show significance without priming.*

Although priming does not introduce false positives, it may cause false negatives: test values which would have appeared just above the significance threshold but which were reduced below the threshold by the effect of priming (increasing denominator of the statistic). This reduction in detail is a natural consequence of the high-privacy context. However, we note that the effect diminishes as the data-set increases: the impact of priming on the value of the test statistic goes to zero as the size of the difference-set goes to infinity.

[2] Inserting an element into the bottom of the difference set naturally increases the rank of all the other, larger elements by 1

Theorem 6.5 $\lim_{N_R \to \inf} \left[\frac{W-0.5}{\sigma(N_R)} - \frac{W-0.5}{\sigma(N_R+2k)} = 0\right]$

Proof:

$\lim_{N_R \to \inf} \left[\frac{W-0.5}{\sigma(N_R)} - \frac{W-0.5}{\sigma(N_R+2k)}\right] =$

$\lim_{N_R \to \inf} \left[(W-0.5)\left(\frac{1}{\sigma(N_R)} - \frac{1}{\sigma(N_R+2k)}\right)\right] =$

$\lim_{N_R \to \inf}[(W-0.5)] * [\lim_{N_R \to \inf}(\frac{1}{\sigma(N_R)}) - \lim_{N_R \to \inf}(\frac{1}{\sigma(N_R+2k)})] =$

$\lim_{N_R \to \inf}[(W-0.5)] * [0-0] =$

$\lim_{N_R \to \inf}[(W-0.5)] * 0 = 0$

6.3.3 Recalibrating the Critical Value Table

Now we address the critical value table. In a non-privatized context, this table is used to determine whether a given test statistic value is sufficiently large to indicate a fundamental difference in the distributions being compared, rather than spurious sampling error. For values of $N_R > 10$, the WSRT statistic can be compared against the same critical value table that belongs to the more widely-used T-Test over normal distributions (see Figure 6.3). However, in addition to sampling error, we will now need to account for error introduced by our privatization noise. We will modify the critical value table such that added noise does not introduce unexpected false-positives (statistics which appear significant solely due to the effect of added privatization noise).

For simplicity, we will use $\epsilon = 1$ as our privacy parameter for the remainder of this chapter. We begin by deriving the following two results relevant to recalibrating the critical value table.

Lemma 6.6 *Given a positive privatized value X, which was privatized by the addition of noise taken from the laplacian(W) distribution: The probability that the true value V, before noise addition, was greater or equal to positive threshold T is: $(1/2)e^{\frac{X-T}{W}}$.*

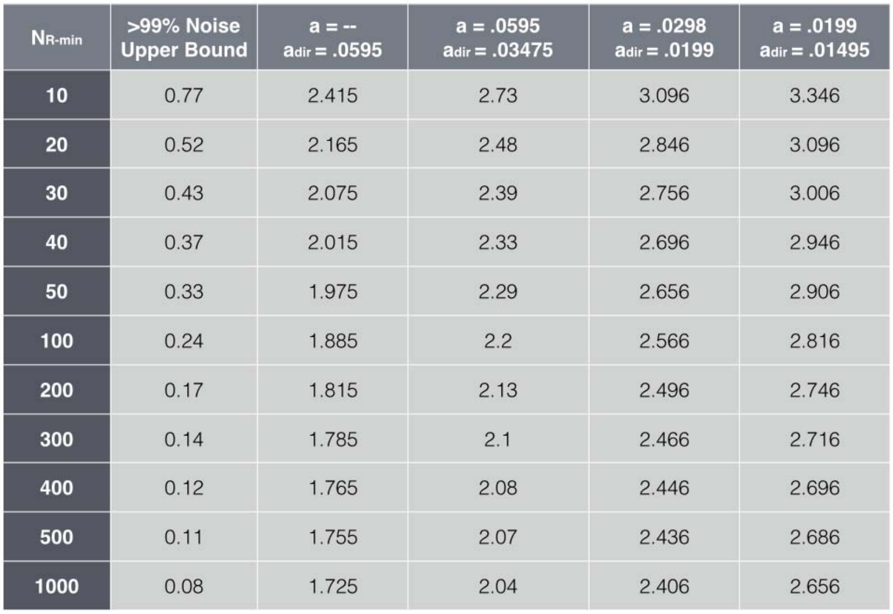
Proof: *Note that: $X = V + noise(W)$, and thus if $noise(W) < (X - T)$, then $V > T$. The CDF for the Laplacian Distribution is: $(1/2)e^{\frac{x-\mu}{b}}$. Plugging in the appropriate values $(\mu = 0, b = W, x = (X - T))$ gives the desired result.*

Theorem 6.7 *Given a privatized test statistic X such that: (1) There is a $t\%$ chance that the true, original value Z was greater than T, and (2) There is a $p\%$ chance that a value of T or greater indicates a significant difference between the two distributions being tested, then the total probability that X indicates a significant difference between the two distributions is $p\% \times t\%$.*

Proof: *This follows trivially from the laws of probability: given two independent events A, B: $P(A \wedge B) = P(A)P(B)$*

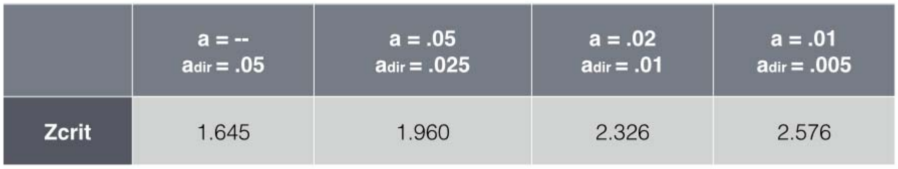
Given Theorem 6.7 and Lemma 6.6, we can now alter the critical value table to take added privatization noise into account (see Figure 6.2). Recall that the magnitude of noise added to the WSRT statistic is dependent on the size of the difference-set. Following Lemma 6.6 we compute for each N_R a 99% effective upper-bound for noise values sampled according to a laplacian distribution with parameter $(\frac{2N_R}{\sigma(N_R)})$; with 99% probability the added noise at these values of N_R will have magnitude smaller than these upper-bounds. We then modify both the a values and the Z_{crit} thresholds in the the traditional critical value table (Figure 6.3) to take into account the effect of this added noise, according to Theorem 6.7.

The revised table is used as follows: The difference-set minimum size indexes the rows, and the probability of significance (revised a-values) indexes the columns. For a given difference-set of size equal to or larger than N_R (with noise added based on a sensitivity of N_R or larger), and significance probability $1 - a$, the value in location (N_R, a) gives appropriate revised Z_{crit} value.



$N_{R\text{-}min}$	>99% Noise Upper Bound	a = -- a_{dir} = .0595	a = .0595 a_{dir} = .03475	a = .0298 a_{dir} = .0199	a = .0199 a_{dir} = .01495
10	0.77	2.415	2.73	3.096	3.346
20	0.52	2.165	2.48	2.846	3.096
30	0.43	2.075	2.39	2.756	3.006
40	0.37	2.015	2.33	2.696	2.946
50	0.33	1.975	2.29	2.656	2.906
100	0.24	1.885	2.2	2.566	2.816
200	0.17	1.815	2.13	2.496	2.746
300	0.14	1.785	2.1	2.466	2.716
400	0.12	1.765	2.08	2.446	2.696
500	0.11	1.755	2.07	2.436	2.686
1000	0.08	1.725	2.04	2.406	2.656

Fig. 6.2.: Recalibrated critical value table. For each of the given N_{R-min} values, the 99%-effective upper bound for sampled noise values is listed, along with adjusted a values (for both directed and undirected hypotheses) and Z_{crit} thresholds.



	a = -- a_{dir} = .05	a = .05 a_{dir} = .025	a = .02 a_{dir} = .01	a = .01 a_{dir} = .005
Zcrit	1.645	1.960	2.326	2.576

Fig. 6.3.: Original (non-privatized) Critical Value Table

6.3.4 Complete Algorithm

We now summarize the complete privatized Wilcoxon Signed-Rank Test procedure as described in the previous sections.

[1a] High-Privacy Prime the difference-set with $2k$ values to achieve a minimum difference-set size $N_R \geq N_{R-min} = 2k$ (see Section 6.3.2).

[1b] **High-Utility** Assert that at least 30% of the data-set of size N has non-zero difference values, producing a minimum difference-set size $N_R \geq N_{R-min} > .3N$

[2] Compute the true Wilcoxon Signed Rank Statistic Z (see the procedure in Section 6.2.1).

[3] Sample a noise value from the $laplacian(\frac{2N_{R-min}}{\sigma(2N_{R-min})})$ distribution, and add this to Z (this assumes $\epsilon = 1$). This produces a publishable value $Z_{priv} = Z + noise$.

[4] Compare the resulting privatized statistic Z_{priv} for significance, against the revised Z_{crit} table (see Figure 6.2).

6.4 Practical Application

Finally, we demonstrate the application of the privatized Wilcoxon Signed-Rank Test to several queries over the New York City Taxi Data-set [65]. This simply-anonymized data-set was originally collected in 2013 and contains very detailed information including, for every cab driver, a log of all trips: trip origin, destination, date, time, fare, number of passengers, distance, and tips. The data-set was publicly released as the result of a Freedom Of Information Law request in 2014, and has been shown to be extremely susceptible to de-anonymization attacks [66]. However, because this data-set has already been made publicly available, we will include both privatized and non-privatized results for comparison purposes.

To demonstrate the need for a privatized statistical significance test, in cases where underlying patterns in the data may not be self-evident, we also include privatized means for each of the three queries. Under variant 2 sensitivity (with fixed N), a differentially privatized mean may be computed as follows:

Definition 6.8 ***Privatized Average:*** *Given a data-set $D = \{d_i\}$ of size n, with upper bound b (such that $d_i \leq b \forall i$), we define the* ***privatized average*** *(under neighboring world variant two) as:*

$$Avg_{priv} = (\frac{1}{n}\sum d_i) + \frac{laplacian(b/\epsilon)}{n}$$

6.4.1 Data-set

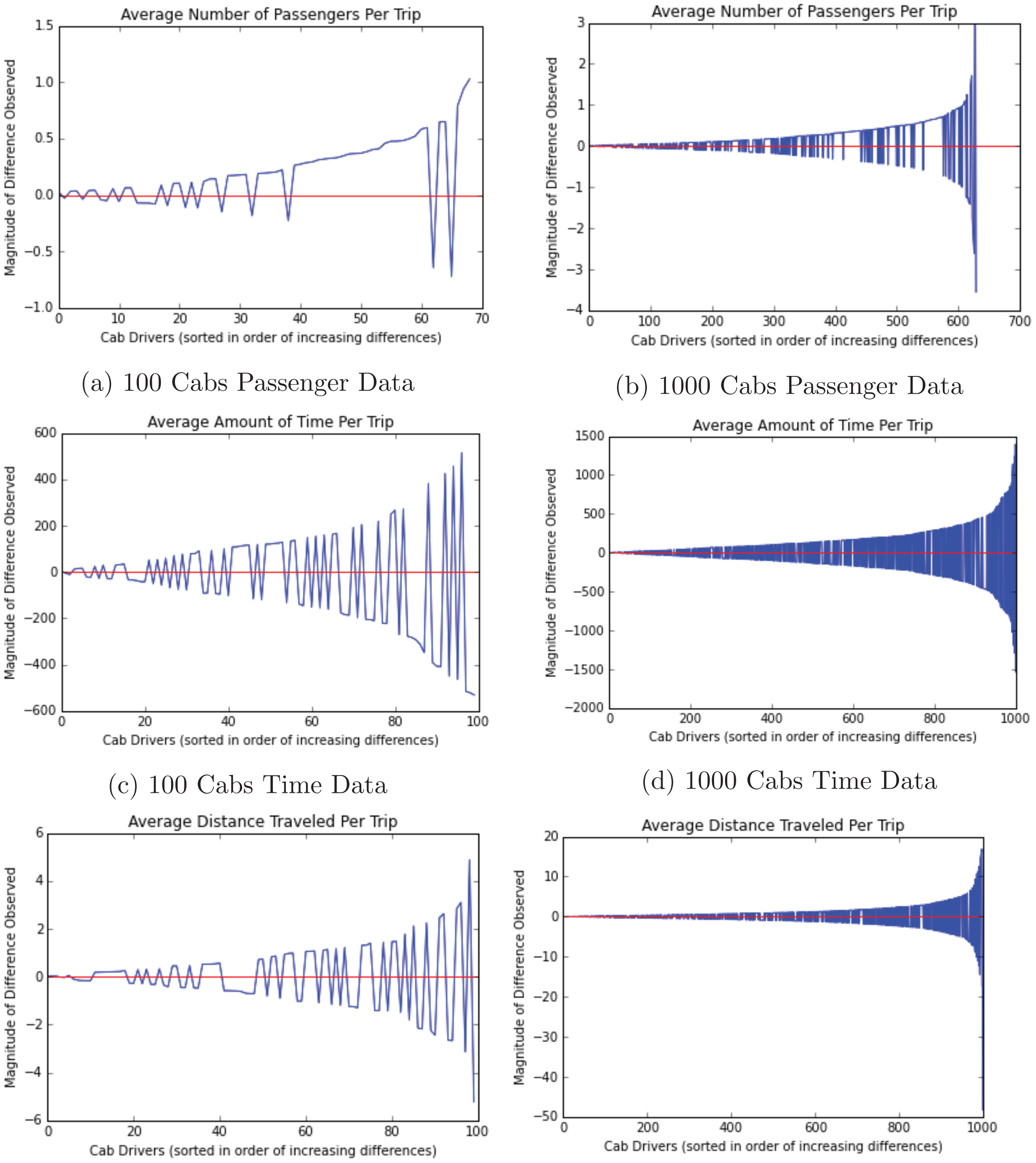
We look at two days from the data-set, January 1st (New Years Day, including trips taken after midnight on New Years Eve) and January 2nd (a Wednesday). A total of 17,069 cab drivers drove on both of these days. For the cab drivers in our data-set we consider how three pieces of data differ between these days:

Queries (Change comparison from 1/1/2013 to 1/2/2013)

- **Car-pooling:** Average number of passengers per trip
- **Duration:** Average time per trip (in seconds)
- **Distance:** Average distance per trip (in miles)

We will protect the privacy of the cab-drivers whose data was contributed to the set. Note that although this data-set was originally presented in tabular form, it has a natural network interpretation as a graph of cab trips (multi-edges) between locations (nodes) in New York City. In this interpretation, the data contributed by each cab driver forms a subgraph of the network, consisting of the trips taken by that driver. The queries we consider above reference edge-properties in this graph.

To demonstrate the impact of data-set size on privatization and significance estimation, we will run each query over a small data set consisting of 100 drivers, and a large data-set consisting of 1000 drivers. Figure 6.4 shows the raw difference-set data for our three queries across the two data-set sizes.



(a) 100 Cabs Passenger Data

(b) 1000 Cabs Passenger Data

(c) 100 Cabs Time Data

(d) 1000 Cabs Time Data

(e) 100 Cabs Distance Data

(f) 1000 Cabs Distance Data

Fig. 6.4.: Sorted raw distance-sets for each of our three queries. Without statistical analysis, it is difficult to draw meaningful conclusions about this data.

6.4.2 Privatized Analysis

For each query and each data-set size, we computed the following statistics:

Statistics

- True mean of the difference-set
- Privatized mean of the difference-set
- True Wilcoxon Signed-Rank Test statistic Z
- Privatized Wilcoxon Signed-Rank Test statistic (stating $N_R \geq .3N$)
- Primed, Privatized Wilcoxon Signed-Rank Test statistic (using a priming set of size 2×15 to produce $N_R \geq 30$)

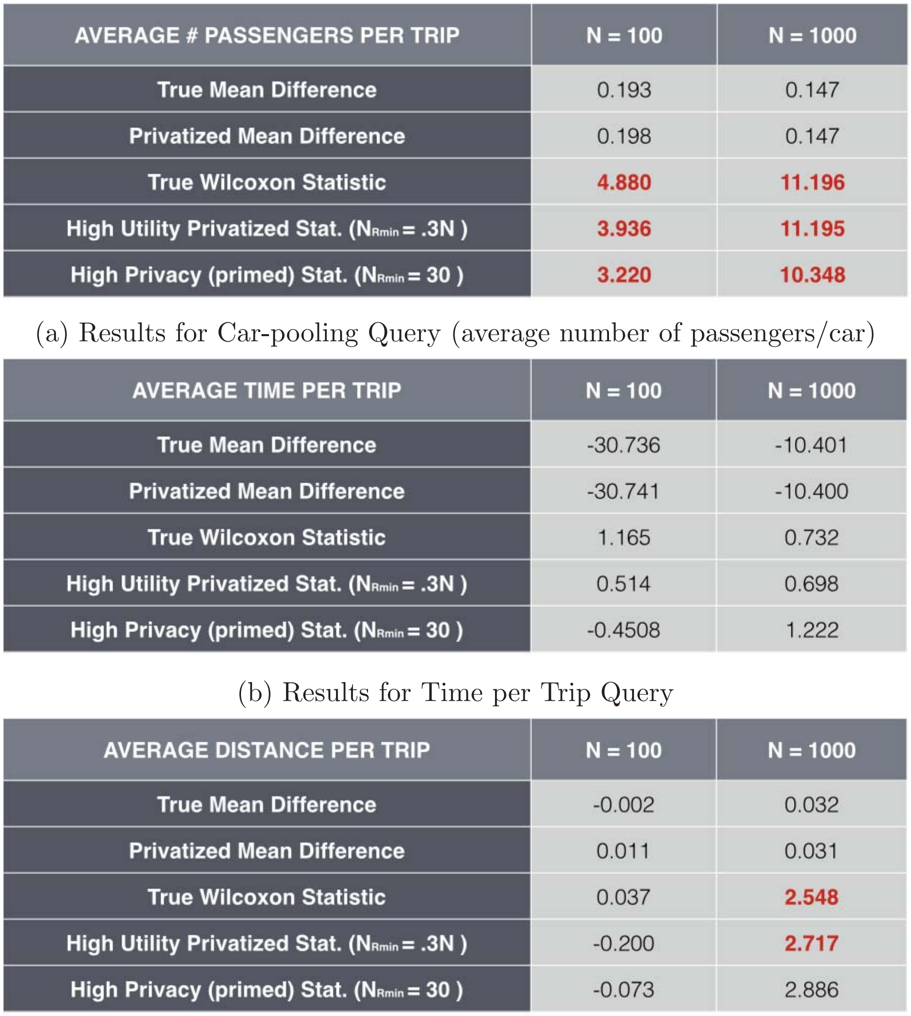
For a significance threshold, we used $a = .02$ for the non-privatized statistics, and $a = 0.0199$ (from the recalibrated significance table, Figure 6.2) for the privatized values, placing a very slightly more rigorous requirement on the privatized computations. Results can be seen in Figure 6.5 with significant statistics highlighted in red.

In this analysis set, each query was computed independently, without summing total sensitivity as in the previous chapters; in practice, publishing this complete analysis set would provide reduced $\epsilon/6$ protection for each individual cab driver. However, there is an alternative: In this context it is possible to run each query over a distinct (disjoint) sample of cab drivers, such that no cab driver contributes to more than one analysis. Our total analysis set requires data from 3,300 cab drivers, while our data-set over these two days contains over 17,000 cab drivers, allowing space for further analyses if desired, without increasing sensitivity.

We are using $\epsilon = 1$ for simplicity of calculation. This is slightly larger than our previous default value of $\epsilon = ln(2)$, and thus will provide a slightly weaker privacy guarantee (see Definition 2.2).

6.4.3 Discussion of Results

First, we note that the mean difference values (both privatized and non) are not very informative. The largest mean difference occurs in the duration query (with



AVERAGE # PASSENGERS PER TRIP	N = 100	N = 1000
True Mean Difference	0.193	0.147
Privatized Mean Difference	0.198	0.147
True Wilcoxon Statistic	**4.880**	**11.196**
High Utility Privatized Stat. (N_{Rmin} = .3N)	**3.936**	**11.195**
High Privacy (primed) Stat. (N_{Rmin} = 30)	**3.220**	**10.348**

(a) Results for Car-pooling Query (average number of passengers/car)

AVERAGE TIME PER TRIP	N = 100	N = 1000
True Mean Difference	-30.736	-10.401
Privatized Mean Difference	-30.741	-10.400
True Wilcoxon Statistic	1.165	0.732
High Utility Privatized Stat. (N_{Rmin} = .3N)	0.514	0.698
High Privacy (primed) Stat. (N_{Rmin} = 30)	-0.4508	1.222

(b) Results for Time per Trip Query

AVERAGE DISTANCE PER TRIP	N = 100	N = 1000
True Mean Difference	-0.002	0.032
Privatized Mean Difference	0.011	0.031
True Wilcoxon Statistic	0.037	**2.548**
High Utility Privatized Stat. (N_{Rmin} = .3N)	-0.200	**2.717**
High Privacy (primed) Stat. (N_{Rmin} = 30)	-0.073	2.886

(c) Results for Distance per Trip Query

Fig. 6.5.: Results of the statistical analysis for our three queries. Highlighted values indicate significance.

drivers on January 2nd having, on average, average trip durations 30 seconds shorter than they experienced on January 1st). However, the distributions for the two dates are not statistically significantly different. By contrast, the car-pooling query has a very small mean difference value, but a significant difference between the distributions overall. Recall that mean computations may be influenced by a few large outlying values that are not characteristic of the underlying distribution; this effect is negated by the signed-rank sum strategy used in the WRST. Privatized means cannot replace the functionality of privatized statistical significance tests.

Returning to the significance results themselves, we see clear evidence of a significant difference in car-pooling across the two dates: Individuals were more likely to share a cab on New Years Day (and returning home after midnight, from celebrations on New Years Eve). This seems reasonably intuitive: friends might share a cab home after a party or event; tourists might share a cab to the airport as they return home on New Year's Day. We see that privatization noise does not effect the significance computation in this case. For the analyses with $N_{R-min} = 30$ (the high-utility small data-set and the two high-privacy sets) the critical threshold for undirected hypotheses with $a = 0.0199$ is 3.006. For the large high-utility data-set, which has $N_{R-min} = .3 \times 1000 = 300$, the significance threshold is . Both the primed and non-primed privatized analyses show significance on both sizes of data-set.

By contrast, we see that the duration of trips did not vary significantly between the two dates (possibly reflecting similar traffic conditions). Due to the very small Z value, the effect of priming, and negative added noise–the smaller primed statistic has a negative value: this can be set as zero before public release of the results, to reduce confusion.

Finally, the results pertaining to distance are interesting. We do not see a significant difference in distances on the smaller data-set, but the larger data-set offers enough detail to see a small significant difference: the true statistic falls above the 2.326 threshold necessary to indicate significance for an undirected hypothesis with $a = 0.02$, and the high-utility privatized statistic falls just above the threshold for $N_{R-min} = 300$, which is 2.716. However, the high-privacy statistic, with a smaller $N_{R-min} = 30$ does not fall above its critical threshold 3.006. This is an example of the higher-privacy context introducing a false negative result. On large data-sets, the choice of a somewhat larger priming set will increase N_{R-min} and reduce the magnitude of added noise; this will reduce the likelihood of false negatives somewhat.

However, the privatized WSRT should not be used for conclusively accepting the null-hypothesis when results are near the significance threshold.

In general, the privatized Wilcoxon Signed-Rank Test provides both a robust privacy guarantee and a robust statistical analysis tool: privatized results which have been shown to be significant are as trustworthy as the original, non-privatized results.

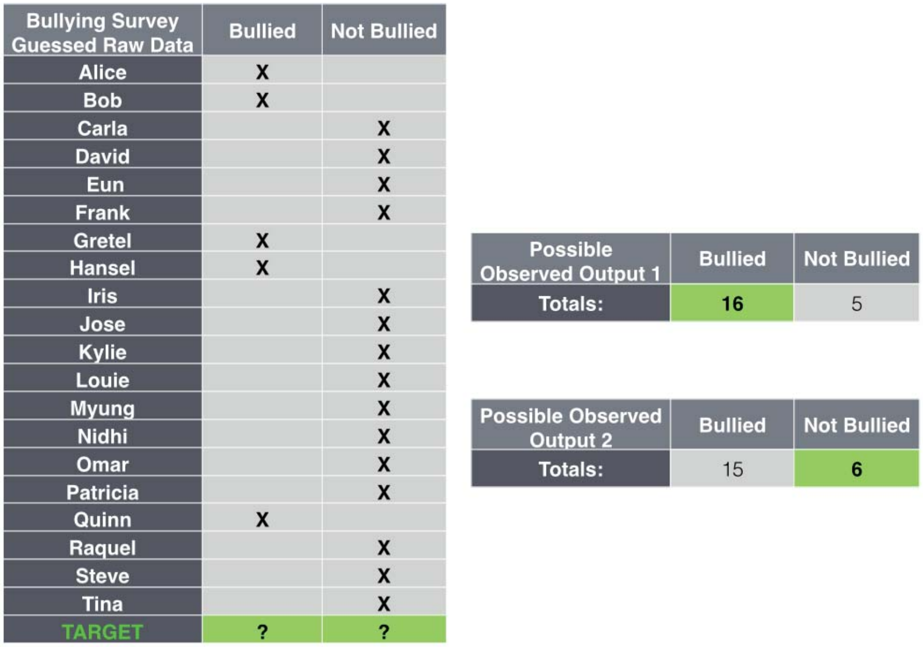
7. DE FACTO PRIVACY

7.1 Introduction

The primary objective of privacy-preserving data mining is to untangle aggregate facts about a population of interest from specific, sensitive facts attached to particular individuals. In Chapter 4 and Chapter 5, we demonstrated three factors that effect differentially private data-mining with Laplacian noise: the sensitivity of the function, the size of the output data-structure, and the size of the data-set. We showed that given a low sensitivity function, a small output space, and a large amount of data, the effect of the noise was nearly undetectable. This begs the question of whether noise addition is necessary at all in these conditions.

We note that added noise is not necessarily the only factor obfuscating a target individual's sensitive contribution to a data-set. Consider the following example: A survey on bullying is distributed to students at a school; it is then aggregated into a count of the number of students who reported having been victimized by bullying, and the total count is posted on a school bulletin board to draw attention to the problem. This count does not satisfy differential privacy: given $(n+1)$ total students, an attacker who knows *with certainty* the responses for n students will be able to accurately determine the response for the $(n+1)$th student. However, for a given data-set and aggregation, we want to formally understand *how much* outside information about the data-set is required for an attacker to feel confident that he will be able to learn the data-value of an unknown individual. We will assume a very strong attacker who has at least some knowledge (either certain or guessed) about every individual in the data-set with the exception of a single unknown target person. We are interested in the attacker's beliefs about his own chances of success in uncovering the truth about this unknown person. Our goal is to develop a metric to measure the relative level of 'De Facto Privacy' provided by commonly-used ad-hoc privacy protections such as simple deterministic aggregation.

7.2 Motivating Exmples



Bullying Survey Guessed Raw Data	Bullied	Not Bullied
Alice	X	
Bob	X	
Carla		X
David		X
Eun		X
Frank		X
Gretel	X	
Hansel	X	
Iris		X
Jose		X
Kylie		X
Louie		X
Myung		X
Nidhi		X
Omar		X
Patricia		X
Quinn	X	
Raquel		X
Steve		X
Tina		X
TARGET	?	?

Possible Observed Output 1	Bullied	Not Bullied
Totals:	**16**	5

Possible Observed Output 2	Bullied	Not Bullied
Totals:	15	**6**

Fig. 7.1.: An example of a simple Yes—No survey

Returning to our bullying survey (see Figure 7.1), assume our attacker is familiar with the students in the school and so, with exception of the target individual, the attacker has a series of guesses about the students' likely responses. He is using "outside knowledge" and has not seen the actual submitted survey papers. He does not have certainty: it's possible that a bullied student lied on the survey, or that an apparently safe student is experiencing bullying where the attacker does not witness it. The attacker believes his guesses are true with probability $p_{correct} = 0.9$. He imagines a scenario where the published totals are: $bullied = [6], nonbullied = [15]$.

It appears, from his guesses, that the target individual has been bullied. But, there is another possible explanation for these totals: the target individual was not bullied, and instead an individual the attacker had guessed was safe was instead bullied. This alternate scenario produces output identical to the scenario in which the target individual is bullied, and by the attacker's own estimation, it will occur with probability $[numberofnonbullied] \times p_{correct}^{[N-1]}(1 - p_{correct})$. In our hypothetical example, that amounts to: $15 \times (0.9)^1 9(0.1) \approx 0.20$, (which we can compare to the $0.9^2 0 \approx 0.12$ chance that all of the attackers guesses were correct.)

Similarly, the attacker might imagine the complementary scenario, in which the published totals are: $bullied = [5], nonbullied = [16]$, and the target individual appears not to be bullied. However, an alternate explanation is that the target individual *was, in fact* bullied, and a bullied individual lied on their survey. The attacker believes this will occur with probability *[number of Bullied]* $\times p_{correct}^{[N-1]}(1 - p_{correct})$ $= 5 \times (0.9)^1 9(0.1) = .07$. Because the bullied set is smaller than the nonbullied set, the attacker believes a mistake in this scenario is less likely than in the previous scenario. So, of the two possible mistakes, *[Target appears Bullied → Target is Nonbullied]* and *[Target appears Nonbullied → Target is Bullied]*, the least likely mistake has probability .07 relative to a fully correct guess. In general, the probability of mistake *[Target appears X → Target is Y]* is *[number of X]*$\times p_{correct}^{[N-1]}(1 - p_{correct})/p_{correct}^{N}$ relative to a fully correct guess.

We can see the effect of one of our privacy factors in these results: as the data set size increases, and the number of students in both the Bullied and Nonbullied categories increases, the probability of a mistake increases. With a very large dataset an attacker will have very little confidence in his inference about the target. We will consider one more illustrative example before formally introducing our model.

We now increase the output size of the query from two disjoint counts to four disjoint counts. A very sensitive survey collects information about high school students' experiences with two facets of the traditional trio: *Sex* and *Drugs* [1] (see Figure 7.2).

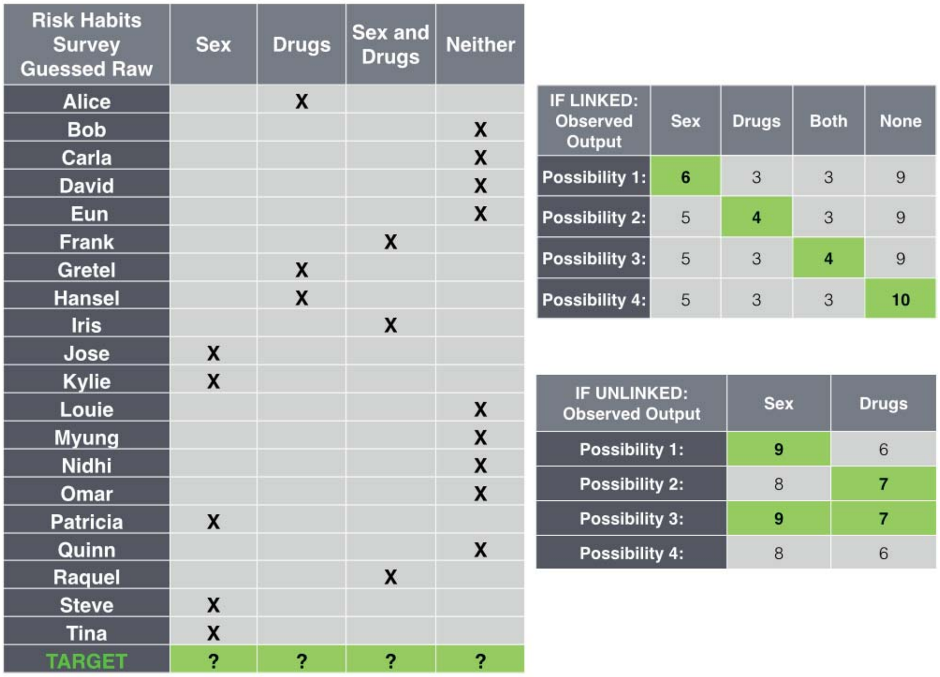
[1] Enjoying Rock-n-Roll music is not as sensitive as it once was.

The data submitted by each individual student falls into one of four categories: Sex, $Drugs$, $Both$($Sex\ And\ Drugs$) or $Neither$. Our first aggregation scheme over this data simply publishes the total counts for each of these four categories. Again, our attacker has a guess for the true value of each student, which he believes to be accurate with probability $p_{correct} = 0.9$. When his guess for a student's data value is incorrect, we make the simplifying assumption that the attacker believes all alternative values are equally probable (if the attacker guesses that $category_J = Sex$, then the attacker believes with probability 0.9 that $category_J = Sex$, and with probability (0.1)(1/3) that $category_J = Neither$, or $category_J = Both$, or $category_J = Drugs$).

There are four possible scenarios when the data is published–the target might appear to be in each of the four categories. However, again, there are possible alternate explanations: For example, if the target appears to be in the *Both* category, it's possible that the target was actually in the $Neither$ category, and an individual the attacker guessed was in the $Neither$ category was, in truth, quietly in the *Both* category. The attacker estimates the probability of a *[Target appears Both → Target is Neither]* mistake to be [*Number of People in Neither*]$\times(0.9)^{n-1}(.01)(1/3) \approx 0.04$]. In our example, this comes to $\approx 0.04/.12$ relative to the probability of a fully correct guess.

There are $4 \times 3 = 12$ total possible swapping mistakes, across all four data publishing scenarios. The least likely mistake, *[Target appears X → Target is Both]*, produces a $3 \times (0.9)^{n-1}(.01)(1/3) \approx .01$ probability for a mistake, relative to the probability of a fully correct guess. In general, *[Target appears X → Target is Y]* mistakes occur with probability *[size of category Y]*$\times(0.9)^{n-1}(.01)(1/3)/(0.9^n)$, relative to the probability of a fully correct guess.

Here we see the impact of another privacy factor, a larger output space. This reduces the attacker's estimated likelihood of any particular mistake by increasing the number of possible alternative categories (introducing the (1/3) factor in the analyses above). It also spreads the data across more categories, reducing the number of individuals in each category (and thus reducing the probability of any guessing-error



Risk Habits Survey Guessed Raw	Sex	Drugs	Sex and Drugs	Neither
Alice		X		
Bob				X
Carla				X
David				X
Eun				X
Frank			X	
Gretel		X		
Hansel		X		
Iris			X	
Jose	X			
Kylie	X			
Louie				X
Myung				X
Nidhi				X
Omar				X
Patricia	X			
Quinn				X
Raquel			X	
Steve	X			
Tina	X			
TARGET	?	?	?	?

IF LINKED: Observed Output	Sex	Drugs	Both	None
Possibility 1:	**6**	3	3	9
Possibility 2:	5	**4**	3	9
Possibility 3:	5	3	**4**	9
Possibility 4:	5	3	3	**10**

IF UNLINKED: Observed Output	Sex	Drugs
Possibility 1:	**9**	6
Possibility 2:	8	**7**
Possibility 3:	**9**	**7**
Possibility 4:	8	6

Fig. 7.2.: An example of a simple two question survey

that involves those categories). A data-set with very many categories and few individuals will tend to have many singleton categories, consisting of uniquely identifiable individuals, and will have many very low-probability mistakes.

In the two previous examples, a mistake of type *[Target appears X* → *Target is Y*, requires that the attacker's guess was incorrect such that *[G was guessed Y* → *G is X]*. In other words, the target must swap places with an incorrectly guessed person. To compute the probability of the mistake, we compute the probability of the complementary incorrect guess. We'll refer to the publication schema demonstrated in these two examples as *radio-button* schemas, adopting the terminology from webform interfaces: individuals answering a radio-button question can choose precisely one option. By contrast, individuals answering a *check-box* question can check as many options as they like, or no options at all. We next discuss a *check-box* schema.

For the second publication scheme on this data, we will reduce the output size by de-linking individual student's responses for the two queries, Sex and Drugs; this produces two counts totaling the number of people who responded Sex and the number of people who responded Drugs. This is a *check-box* publication scheme: Individuals in the category 'Both' will check both boxes and contribute to both counts, and individuals in the category 'Neither' will check no boxes and contribute to neither count.

We now consider the effect of this change on our mistake probability computations. For the mistake *[Target appears Neither $\rightarrow$ Target is Both*, there is the swapping explanation we saw previously: *[G was guessed Both $\rightarrow$ G is Neither]*. However, there are now additional possible guess-errors that may account for the attacker's guess having miscounted one extra D and S. For example: *[G_1 was guessed D and G_2 was guessed S, $\rightarrow$ G_1 and G_2 are Neither]*, and *[G_1 and G_2 were guessed Both, $\rightarrow$ G_1 is S and G_2 is D]*. Because these additional guess-error explanations each have non-zero probability, the total probability for any *[Target appears X $\rightarrow$ Target is Y]* mistake in the check-box schema is greater than or equal to the probability for the same mistake in the radio-button schema over the same raw data-set.

7.3 Defacto Privacy Metric

We now have sufficient background to introduce our formal model and derive a useful set of theoretical results.

7.3.1 Framework

We begin with a raw data set, which we will represent as the data collected over a set of individuals I using a series of questions Q, such that question q_j has a_j possible answer values. Note that it is possible to expand this question set into $k = \Sigma_j a_j$ binary questions of the form: "Did the individual choose answer $a_{j,l}$ to question q_j: (Y/N)?" Considering the complete response submitted by one individual to the k

binary questions, we can see that there are a total of at most 2^k possible data values that an individual might have, or 2^k possible *categories* of individuals. We note that if $|I| < 2^k$, at least one category must be empty. Categories which represent invalid of response-sets to binary questions (for instance, answering ”Yes” to two mutually exclusive questions such as ”GPA = A: (Y/N)”? and ”GPA = B: (Y/N)?”) will also be empty. In general, we will not attempt to distinguish these two cases, implicitly assuming that empty categories represent values of either zero or negligible likelihood.

For all but one individual in I, the attacker has ”guessed” a category value $(guessed(i) = category_j)$[2] These guesses form a guess-set G. The attacker believes each of his guesses to be true with probability $p_{correct}$. If an individual is guessed to be $guessed(i) = category_j$, then for every other $category_x \neq category_j$ the attacker believes $guessed(i) = category_x$ with equal probability $\frac{(1-p_{correct})}{2^{k-1}}$. For the target individual, the attacker has no information.

In addition to his guesses G, the attacker has access to the published information about the raw data-set. This published information is deterministic; it reflects the true data-set with no added noise. We refer to the format of the published information as a *publication schema*, S.

Intuitively, we want to measure the attacker's relative *self-confidence* about his ability to correctly infer $category_{target}$, given guess-set G and schema S, assuming the best-case scenario for the published information: the published information does not contradict the hypothesis that all of the attacker's guesses are correct, so that the target's value appears to be the difference between the published data and the guessed values. Schemas that provide better privacy will decrease the attacker's self-confidence, even in this best-case scenario.

We emphasize that we are not claiming to provide a computation of the attacker's true probability of correctly inferring a target individual's data value (in deterministic settings, we feel that this is dependent on an infeasible number of contextual

[2] In a realistic scenario, it is possible the attacker knows the true values of some portion of the data-set with absolute certainty, because he has been able to get partial access to the true data-set. WLOG, we assume that I is the portion of the data set which is not known with certainty.

factors). Instead, we hope to provide a useful abstraction for characterizing the ways in which publication schemas work to magnify uncertainty. This can be used to make a well-founded comparison of the relative privacy provided by different publication schemas, and it can provide a mathematical framework for more formally understanding common intuitive notions about privacy.

We mathematically abstract the attacker's fear of his own fallibility as follows: Given his guesses G, and a hypothetical set of published information in which the target *appears* to have value $category_t$, the attacker considers mistakes of the form *[Target appears $X \rightarrow$ Target is Y]*. For any hypothetical set of published information, there are 2^{k-1} possible mistakes, producing a total of $2^k \times (2^{k-1})$ possible mistakes total over all possible cases (we refer to this as the mistake-set, M). We compute the probability of a mistake as the combined probability of possible simple guess-errors which could cause the mistake to occur. To normalize for data-set size, we consider the probability of a mistake relative to the probability of a fully correct guess-set, ($p^N_{correct}$). Note that the probability of a fully correct guess-set is constant for a given data-set, the probability of a mistake is dependent on the guessed distribution of the data across the possible categories (G), the size of the question set (k), and the publication schema (S).

Schemas

In this chapter we will consider two basic publication schemas:

Definition 7.1 ***Radio-Button Schema:*** *A Radio-Button Schema publishes information about the data-set by listing the total counts in each of the 2^k possible categories that have non-zero counts (note that if the data-set does not include individual names, this is functionally equivalent to publishing a simply-anonymized data-set).*

Definition 7.2 ***Check-Box Schema:*** *A Check-Box Publication Schema publishes information about the data-set by listing the total counts for each of the k binary questions. One individual can affect up to k of these counts.*

7.3.2 Theoretical Results

We now present a series of theoretical results regarding radio-button and checkbox schemas. We begin with a formal summary of several facts about Radio-Button schemas that were observed in our motivating examples. We consider the effect of the schema on the probability of every mistake in the mistake-set M (all $2^k \times (2^{k-1})$ mistakes of the form *[Target appears X* $\rightarrow$ *Target is Y]*).

Theorem 7.3 *In the Radio-Button schema, the probability of mistake* [Target appears X $\rightarrow$ Target is Y] *is* $|category_Y| \times (p_{correct})^{n-1}(.01)(\frac{1}{2^{k-1}})$

Proof: *In the Radio-Button schema, a mistake* [Target appears X $\rightarrow$ Target is Y] *requires a guessing-error* [Target guessed Y $\rightarrow$ Target is X]. *This error requires one incorrect guess in category*$_Y$ *and correct answers for the remainder of the data-set. The probability of an incorrect guess in category*$_Y$ *is* $|category_Y| \times (p_{correct})^{n-1}(.01)$*, and the probability that the incorrectly guessed person is actually in category*$_X$ *is* $\frac{1}{2^{k-1}}$. *Thus the total probability of this mistake is as stated,* $|category_Y| \times (p_{correct})^{n-1}(.01)(\frac{1}{2^{k-1}})$

Corollary 7.4 *In the Radio-Button Schema, increasing the size of the data-set will monotonically increase the probability of the mistakes in* M *(if* $|I_1| \geq |I_2|$*, then* $\forall m in M$*,* $prob(m|I_1) \geq prob(m|I_2)$ *)*

Proof: *Adding any individual* i *to the data-set will increase the size of* $category_{guessed(i)}$. *This will increase the probability of mistakes of the form* [Target appears X $\rightarrow$ Target is $category_{guessed(i)}$]. *Because adding an individual will not decrease the size of any category and will not increase the number of binary questions, it will not decrease the probability of any mistake. By induction, adding any set of individuals* $\{i\}$ *will monotonically increase the probability of mistakes in* M *as described.*

Theorem 7.5 *In the Radio-Button Schema, increasing the number of binary questions (by adding either additional questions or additional possible answer values) will decrease both the average probability of mistakes in M.*

Proof: *Increasing the value of k by 1 by including the question: "Does i have property A (T/F)?", will have the effect of splitting each mistake* [Target appears X $\rightarrow$ Target is Y] *into four cases:* [Target appears X_T $\rightarrow$ Target is Y_T], [Target appears X_T $\rightarrow$ Target is Y_F], [Target appears X_F $\rightarrow$ Target is Y_T], *and* [Target appears X_F $\rightarrow$ Target is Y_F].

Note that $|category_Y| = |category_{Y_F}| + |category_{Y_T}|$*, and thus* $|category_{Y_F}| \leq |category_Y|$ *and* $|category_{Y_F}| \leq |category_Y|$*. We simplify notation by defining* $probError(n, k) = (p_{correct})^{n-1}(.01)(\frac{1}{2^{(k+1)-1}})$*. Note that* $probError(n, (k+1)) = (1/2)probError(n, k)$*.*

Thus, the probability of any mistake in our new set of four mistakes (for example, [Target appears X_T $\rightarrow$ Target is Y_F]*, whose probability is* $|category_{Y_F}| \times probError(n, k+1)$*) will be less than or equal to the probability of our original mistake* $|category_Y| \times probError(n, k)$*, with equality holding only when* $|category_Y| = 0$*. And the average probability of these four mistakes will necessarily be smaller than the probability of the original mistake. Expanding the result over the entire mistake set, we see that if the data-set is non-empty, then the overall average mistake probability is decreased by the addition of any binary question, and by induction any increase to the question-set will decrease the average mistake probability.* [3]

We now look at the relationship between the Check-Box schema and the Radio-Button schema.

Theorem 7.6 *For each mistake $m \in M$, a Check-Box schema will produce a mistake probability greater than or equal to that produced by a Radio-Button schema over the same data-set.*

[3] Interestingly, it can be shown that the sum of the the probabilities in the mistake-set is unchanged

Proof: *Given a mistake* [Target appears X → Target is Y]*, a guessing-error of type* [Guessed($category_i$) = Y → $category_i$ is X] *will account for this mistake in both the Radio-Button and the Check-Box schemas; this establishes equality. However, the CheckBox schema can introduce additional guessing-error explanations for many mistakes.*

We will formally describe one class of introduced guessing-errors: For a given category Y, we will use T_Y to refer to the set of binary questions answered positively. Consider $T_{Y-part} = \{t_1, t_2...t_m\}$ to be any partitioning of T_Y into subsets. Note that for each of these subsets t_j there exists a category X_j such that $T_{X_j} = t_j$. If possible (if all categories X_j that are referenced in the partition are non-empty), choose an arbitrary individual $i_j \in X_j$ for each X_j. Then the increased guess-error set includes $[\bigwedge_j (guessed(category(i_j)) = X_j) \rightarrow \forall i_j, category(i_j)\ isX_\emptyset]$, where $X_\emptyset$ is the category such that $T_{X_\emptyset} = 0$. All combinations of individuals in all non-empty partitioning schemes for T will introduce new possible guess-errors for this mistake.

We note with interest that our third privacy factor, sensitivity, behaves differently in our model than in the differentially private analyses. Although the Check-Box schema increases the sensitivity in comparison to the Radio-Button schema (by increasing the number of published values that one individual can contribute to), it also increases the number of possible interpretations for any observed pattern in the data. This effect isn't apparent in sensitivity costs, which are computed using worst-case hypothetical data-sets and an implicit assumption that an attacker knows with certainty the values for $n - 1$ individuals. Differential privacy gives a robust, absolute guarantee of individual privacy, but our De Facto model is able to capture a few interesting properties that emerge in less extreme cases.

We now look at an ad hoc privacy measure that is often used in real world deterministic publication schemas: grouping together distinct attribute values into a single joint value. For example, in the Check-Box schema from our Sex/Drugs survey example, we might choose to instead group the two separate S, D counts into a single $Risk = S \vee D$ count that would simply count the total number of students who fell into the group $category_{Sex} \cup category_{Drugs} \cup category_{Both}$. Intuitively this increases the privacy of the students to some degree; we can observe this effect formally in our model.

Corollary 7.7 *In a Radio-Button schema, grouping sets of categories (replacing a set of categories $\{C_1, C_2, ...C_j\}$ with one category $C_{Group} = \bigcup C_i$) will monotonically increase the average probability of mistakes.* [4]

Proof: *This is in fact the inverse operation of the splitting procedure (by inclusion of additional binary questions) discussed in Theorem 7.5. Removing categories is equivalent to removing binary-questions, thus this result follows from the previous Theorem.*

Corollary 7.8 *A degree distribution with a cut-off and a published count of individuals falling above the cut-off, is an example of a Grouped Radio-Button schema.*

Proof: *If the raw data-set is the social network (the guess-set G consists of friendship lists for each individual in the network and a category is a specific set of friends), the count for degree d in the the degree distribution is the count of a joint category that groups together all friends-list categories of size d. A cut-off with a published count is equivalent to grouping all individuals in the categories that fall above the cut-off. Other network distributions we have discussed in previous chapters (such as LCC distributions) will behave analogously. DeFacto privacy in these distributions depends on the amount of data, the number of small count categories (which will produce low-probability mistakes), and number of histogram buckets (which determines the number of binary questions asked).*

We also note that results that pertain to Radio-Button schemas have implications for Check-Box schemas as well:

[4] Note that Grouping is distinct from a Check-Box Schema in that a Grouped Radio-Button schema is still a partitioning of the data-set: no individual can contribute to more than one group.

Corollary 7.9 *Both Grouping and increasing the size of the data-set will increase the average probability of mistakes in Check-Box schemas.*

Proof: *This follows from our previous results. Since both Grouping and increasing the data-set size increase the base probability of mistakes in the foundational Radio-Button schema (Corollary 7.7 and Corollary 7.4 respectively), and the Check-Box schema preserves those increased probabilities in addition to adding new possible explanations to the guess-error sets (Theorem 7.6), Check-Box schema mistake probabilities will also be increased by these steps.*

Finally, we will briefly discuss the relationship between low-probability mistakes and factors that have been observed to increase the likelihood of re-identification in real world data-sets. Looking at the mistake probabilities in the Radio-Button schema (Theorem 7.3), we see that in a data-set of size n, a very low (but non-zero) probability mistake *[Target appears $X \rightarrow$ Target is Y]* arises from two factors:

- The size of category Y is small (possibly even a singleton). This means that individuals in this category are rare and may stand out in the data.
- The question set is large (k is large). This means that more information is collected about each individual. This additional information may make it easier to identify an individual in the data-set (even with less complete prior guesses than we assume in De Facto privacy).

An individual i who is unique in the data-set and who has a large set of attributes will introduce a set of very low probability mistakes *[Target appears $X \rightarrow$ Target is category$_i$]*. A data-set with a large number of low-probability possible mistakes is a concern for privacy.

Relevant to this, we note the following result with respect to k-anonymity:

Theorem 7.10 *In the Radio-Button Schema, enforcing k-anonymity over* all *attributes of a data-set with q binary questions will result in a lower-bound of*

$k(p_{correct})^{n-1}(.01)(\frac{1}{2^{q-1}})$ on mistake probability. However, only enforcing k-anonymity over a subset of 'quasi-identifying' attributes will not provide this lower-bound on mistake probability.

Proof: *Following from Theorem 7.3, we know that the probability of a mistake* [Target appears X → Target is Y] *is $|category_Y| \times (p_{correct})^{n-1}(.01)(\frac{1}{2^{k-1}})$. If all categories have a minimum size of k, the result follows. However, if only a subset of 'quasi-identifying' attributes are considered in the k-anonymity rule, then singleton categories (split by attributes that are considered not to be quasi-identifying) are still possible, and low-probability mistakes may exist.*[5]

7.4 Practical Example

We will now briefly illustrate the De Facto model's interpretation of a real world controversy over deterministic data publication.

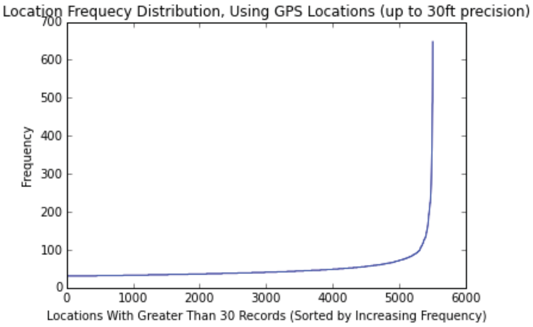


Fig. 7.3.: Distribution of frequent GPS locations across all trips on 1/1/2013-1/2/2013

In Chapter 6, we referenced the New York City Taxi Data set [65]. This data was collected in 2013, published in a simply-anonymized data-set in response to a Freedom of Information Law request in 2014, and was de-anonymized very shortly

[5] Recall that l-diversity ensures that there are several different non-quasi-identifying attribute values appearing in each quasi-identifying category; it does not ensure that every possible category of individual (ie, considering all attributes) is well-populated.

thereafter [66]. As a result of the de-anonymization, sensitive data such as the tipping habits and evening destinations of celebrities became public knowledge. The data-set included (among other data), for every cab driver, a log of all trips: trip origin and destination (in GPS coordinates accurate to 1 meter), date, time, fare, number of passengers, distance, and tips.

In late 2014, taxi competitor Uber offered to release their own New York City data-set, in what they felt was a more privacy-preserving format [67]. Their proposed data-set would consist of a set of independent trips, including the time of the trip and the origin and destination as zip codes. There would be no information on cab driver which could be used to link trips.

In the original NYC taxi data-set there are over 20,000 unique drivers with an average of approximately 20 trips per day. The distribution of locations (within 10m precision) that were visited at least 30 times over January 1st and 2nd is given in Figure 7.3; the vast majority of locations were visited less than 30 times.

Under the De Facto model, the NYC Taxi data set is a Radio-Button schema with an incredibly large question set. Consider the number of categories present in one day's data: Assuming there are d distinct drivable GPS locations in NYC (within a granularity of 1 meter), then there are d^2 possible trips, and there are approximately d^{40} possible series of 20 trips. Assuming cabs generally take 1-3 passengers, there are $d^{40} \times 3^{20}$ possible trip series with car-pooling information included. Adding miles traveled (which depends on route), departure and arrival times, and tip amount, results in a data-set in which essentially every category is a singleton and k is extremely large. We would expect this data-set to have a high-probability of de-anonymization.

Alternatively, with respect to the Taxi data, the Uber proposal introduced both Grouping (by zip code, car-pool, and tip), and a Check-Box schema (in which a cab driver's trips were de-linked). This would significantly decrease the number of low-probability mistakes by decreasing the number of categories (and thus the total number of mistakes), and increasing the probability of any given mistake (by increasing the set of guess-errors which could induce a given mistake). Unfortunately, Uber's proposal was denied and Uber has been required by the New York City government to submit a data-set in the same format as the 2013 Taxi data-set; the full Uber data may now also be accessible by Freedom of Information Law requests [68].

7.5 Summary

The De Facto model provides a method for estimating the degree to which a given data-set and publication schema magnify an attacker's uncertainty about the population, inhibiting his ability to use outside information to correct infer the true value of a target individual; we demonstrated that, in the Radio-Button schema, a very large and evenly distributed data-set with a very small question-set (small value of k) requires the attacker to possess a potentially infeasible degree of certainty about all of the individuals in the data-set in order to have a high probability of correctly uncovering their target individual. We have additionally demonstrated that our De Facto model provides a mathematical framework that captures the following intuitive ideas about privacy in deterministic settings:

- Privacy increases as the data-set size increases (Theorems 7.4 and 7.9)
- Privacy decreases as the output space (and the amount of information collected about each individual) increases (Theorem 7.5).
- Privacy increases as attribute precision is reduced. (Theorems 7.7 and 7.9)
- Privacy increases as records are de-linked. (Theorem 7.6)
- K-anonymity provides privacy protection, but the effectiveness of this protection is reduced if attributes are designated as not quasi-identifying. (Theorem 7.10)

8. CONCLUSIONS

This work proposed to address the question: ”Is practically usable, privacy-preserving social network analysis feasible?” We began by stating several properties that are important for a privatization technique to be practically usable in real world contexts:

- **Guaranteed Privacy**: It must provide a well-defined privacy guarantee to individuals in the data-set.
- **Maintain Utility**: It must enable privatized analyses to produce results with a reasonable level of accuracy.
- **Practically Adoptable**: To encourage adoption it must not impose a significant burden in computing power or mathematical expertise in comparison to the non-privatized analysis it replaces.

We described existing work in simple anonymity, k-anonymity, differential edge-privacy and node-privacy which did not concurrently achieve all three goals (Chapter 3). We then introduced two new adaptations of differential privacy to social network data: Contributor-Privacy (Definition 4.1), which protects the information each individual contributes to the analysis, and Partition-Privacy (Definition 5.1), which protects entire disjoint subgraphs.

These new adaptations of differential privacy enabled us to design privacy-preserving social network techniques which provide robust guarantees of individual privacy while producing high utility results. We demonstrated the ability of our privatized approach to easily and safely gather information for the following network analyses:

Privacy-preserving Social Network Analysis

- Local Clustering Coefficients (with node-level information) [Section 4.4.1]
- Degree Distributions (with node-level information) [Section 4.4.1]
- Reciprocity [Section 4.4.2]
- Homophily [Section 4.2.3]
- Edge Properties [Section 4.2.3]
- Centrality/Community Structure [Section 4.2.4]
- Degree Distributions (with subgraph-level information) [Section 5.2.2]
- Local Clustering Coefficients (with subgraph-level information) [Section 5.4.2]
- Average Shortest Path Length [Section 5.4.2]
- Edge Density [Section 5.4.2]
- Community Counts [Section 5.4.2]

Additionally, to ensure that privatized analysis provides the level of rigor required for social science research (especially research that may be used to inform policy decisions), we introduced a method of determining statistical significance for paired samples under differential privacy using the Wilcoxon Signed-Rank Test, which is appropriate for non-normally distributed data such as social network analysis metrics. (Chapter 6)

This work provides a significant body of evidence to support the claim that our original question can be answered affirmatively: Practically usable, privacy-preserving social network analysis is feasible, in many cases. This result is due both to our novel adaptations of differential privacy to network data, and to our design of high-utility privatized distributions for network analysis.

In our final contribution, we looked one step further to consider the mechanisms that enable our privatized distributions to provide some level of privacy protection, even before the addition of noise required to achieve differential privacy. We defined the De Facto privacy model for formally comparing the relative privacy of deterministic data publication schemas, and proved results related to two schemas, the Radio-Button and Check-Box schemas. (Chapter 7) We demonstrated that our choice of distributions for publishing social network data contributes significantly to the privacy protection offered by our analyses (Theorem 7.8).

We hope that this foundational work will provide future social network analysts with an array of possible options for easily, effectively, and *safely* analyzing, sharing and publishing sensitive social network data.

LIST OF REFERENCES

LIST OF REFERENCES

[1] A. Martínez, Y. Dimitriadis, B. Rubia, E. Gómez, and P. de la Fuente, "Combining qualitative evaluation and social network analysis for the study of classroom social interactions," *Computer Education*, vol. 41, no. 4, pp. 353–368.

[2] A. Engel, C. Coll, and A. Bustos, "Distributed teaching presence and communicative patterns in asynchronous learning: Name versus reply networks," *Computer Education*.

[3] M. De Laat, V. Lally, L. Lipponen, and R.-J. Simons, "Investigating patterns of interaction in networked learning and computer-supported collaborative learning: A role for social network analysis," *International Journal of Computer-Supported Collaborative Learning*, vol. 2, no. 1, pp. 87–103, 2007.

[4] A. B. Firdausiah Mansur and N. Yusof, "Social learning network analysis model to identify learning patterns using ontology clustering techniques and meaningful learning," *Computer Education*.

[5] C. Fidas, V. Komis, and N. Avouris, "Heterogeneity of learning material in synchronous computer-supported collaborative modelling," *Computer Education*.

[6] X.-Y. Wei and Z.-Q. Yang, "Mining in-class social networks for large-scale pedagogical analysis," in *Proceedings of the 20th ACM International Conference on Multimedia*, MM, pp. 639–648, 2012.

[7] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. Vol. 33, pp. 452–473, 1977.

[8] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, pp. 111–125, 2008.

[9] T. Bloom, *Data Access for the Open Access Literature: PLOSs Data Policy*, December 2013.

[10] M. Garcia-Herranz, E. Moro, M. Cebrian, N. A. Christakis, and J. H. Fowler, "Using friends as sensors to detect global-scale contagious outbreaks," *PLoS ONE*, vol. 9, p. e92413, April 2014.

[11] C. H. C. A. Henning, N. Zarnekow, J. Hedtrich, S. Stark, K. Trk, and M. Laudes, "Identification of direct and indirect social network effects in the pathophysiology of insulin resistance in obese human subjects," *PLoS ONE*, vol. 9, p. e93860, April 2014.

[12] K. E. Arnold and M. D. Pistilli, "Course signals at Purdue: Using learning analytics to increase student success," in *Proceedings of the Second International Conference on Learning Analytics and Knowledge*, LAK, pp. 267–270, 2012.

[13] V. Strauss, *Privacy Goncerns row over Gates-funded Student Database*, June 2013. `http://www.washingtonpost.com/blogs/answer-sheet/wp/2013/06/09/privacy-concerns-grow-over-gates-funded-student-database/?print=1`.

[14] S. Mack, *Putting Student Data To The Test To Identify Struggling Kids*, March 2014. `http://www.npr.org/2014/04/08/300587823/putting-student-data-to-the-test-to-identify-struggling-kids`.

[15] E. Zheleva and L. Getoor, "Privacy in social networks: A survey," in *Social Network Data Analytics*, p. 277, 2011.

[16] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," *30th IEEE Symposium on Security and Privacy*, pp. 173–187, 2009.

[17] T. Zhu, S. Wang, X. Li, Z. Zhou, and R. Zhang, "Structural attack to anonymous graph of social networks," *Mathematical Problems in Engineering*, vol. 2013, 2013.

[18] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography* (S. Halevi and T. Rabin, eds.), vol. 3876 of *Lecture Notes in Computer Science*, pp. 265–284, Springer Berlin Heidelberg, 2006.

[19] J. Blocki, A. Blum, A. Datta, and O. Sheffet, "Differentially private data analysis of social networks via restricted sensitivity," *CoRR*, vol. abs/1208.4586, 2012.

[20] L. Zou, L. Chen, and M. T. Özsu, "K-automorphism: A general framework for privacy-preserving network publication," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 946–957, 2009.

[21] F. Ahmed, R. Jin, and A. X. Liu, "A random matrix approach to differential privacy and structure preserved social network graph publishing," *CoRR*, 2013.

[22] C. Dwork, "Differential privacy: A survey of results," in *Theory and Applications of Models of Computation* (M. Agrawal, D. Du, Z. Duan, and A. Li, eds.), Lecture Notes in Computer Science, pp. 1–19, Springer Berlin / Heidelberg, 2008.

[23] M. Hay, V. Rastogi, G. Miklau, and D. Suciu, "Boosting the accuracy of differentially private histograms through consistency," *Procedings of the VLDB Endowment*, vol. 3, pp. 1021–1032, September 2010.

[24] M. Hay, C. Li, G. Miklau, and D. Jensen, "Accurate estimation of the degree distribution of private networks," *Data Mining, IEEE International Conference on*, pp. 169–178, 2009.

[25] V. Karwa, S. Raskhodnikova, A. Smith, and G. Yaroslavtsev, "Private analysis of graph structure," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, 2011.

[26] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing*, STOC, pp. 75–84, 2007.

[27] P. Marsden, "Network data and measurement," *Annual Review of Sociology*, pp. 435–463, 1990.

[28] D. Garlaschelli and M. I. Loffredo, "Patterns of link reciprocity in directed networks," *Physical Review Letters*, vol. 93, no. 26, p. 268701, 2004.

[29] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, pp. 415–444, 2001.

[30] M. Newman, "The structure and function of complex networks," *SIAM Review*, pp. 167–256, 2003.

[31] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, June 1998.

[32] T. G. Kolda, A. Pinar, T. Plantenga, C. Seshadhri, and C. Task, "Counting triangles in massive graphs with mapreduce," *SIAM J. Scientific Computing*, vol. 36, no. 5, 2014.

[33] P. Holland and S. Leinhardt, "Local structure in social networks," *Sociological Methodology*, vol. 7, no. 1, 1976.

[34] A. Degenne and M. Forsé, *Introducing Social Networks*. SAGE Publications Ltd, 1999.

[35] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.

[36] A. Goldenberg, A. X. Zheng, S. E. Fienberg, and E. M. Airoldi, "A survey of statistical network models," *Foundations and Trends in Machine Learning*, vol. 2, no. 2, pp. 129–233, 2010.

[37] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava, "Anonymizing social networks," *Computer Science Department Faculty Publication Series*, p. 180, 2007.

[38] D. F. Nettleton, D. Sáez-Trumper, and V. Torra, "A comparison of two different types of online social network from a data privacy perspective," in *Modeling Decisions for Artificial Intelligence*, pp. 223–234, 2011.

[39] L. Sweeney, "K-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[40] K. Liu and E. Terzi, "Towards identity anonymization on graphs," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 93–106, ACM, 2008.

[41] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *IEEE 24th International Conference on Data Engineering, ICDE 2008*, pp. 506–515, IEEE, 2008.

[42] K. Stokes and V. Torra, "Reidentification and k-anonymity: a model for disclosure risk in graphs," *Soft Computing*, vol. 16, no. 10, pp. 1657–1670, 2012.

[43] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data, TKDD*, vol. 1, no. 1, p. 3, 2007.

[44] Y. Wang, X. Wu, J. Zhu, and Y. Xiang, “On learning cluster coefficient of private networks,” *Social Network Analysis and Mining*, vol. 3, no. 4, pp. 925–938, 2013.

[45] E. Shen and T. Yu, “Mining frequent graph patterns with differential privacy,” in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD, pp. 545–553, 2013.

[46] V. Karwa and A. Slavkovi, “Differentially private graphical degree sequences and synthetic graphs,” in *Privacy in Statistical Databases* (J. Domingo-Ferrer and I. Tinnirello, eds.), vol. 7556 of *Lecture Notes in Computer Science*, pp. 273–285, Springer Berlin Heidelberg, 2012.

[47] Y. Wang, X. Wu, and L. Wu, “Differential privacy preserving spectral graph analysis,” in *Advances in Knowledge Discovery and Data Mining*, pp. 329–340, Springer Berlin Heidelberg, 2013.

[48] D. J. Mir and R. N. Wright, “A differentially private graph estimator,” in *Proceedings of the 2009 IEEE International Conference on Data Mining Workshops*, pp. 122–129, IEEE Computer Society, 2009.

[49] V. Karwa, A. Slavkovi, and P. Krivitsky, “Differentially private exponential random graphs,” in *Privacy in Statistical Databases*, vol. 8744 of *Lecture Notes in Computer Science*, pp. 143–155, Springer International Publishing, 2014.

[50] D. Proserpio, S. Goldberg, and F. McSherry, “A workflow for differentially-private graph synthesis,” 2012.

[51] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Y. Zhao, “Sharing graphs using differentially private graph models,” in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement*, pp. 81–98, 2011.

[52] A. Gupta, A. Roth, and J. Ullman, “Iterative constructions and private data release,” in *Theory of Cryptography Conference*, pp. 339–356, 2012.

[53] Y. Wang and X. Wu, “Preserving differential privacy in degree-correlation based graph generation,” *Transactions In Data Privacy*, vol. 6, 2013.

[54] A. Marin and B. Wellman, “Social network analysis: An introduction,” *Handbook of Social Network Analysis*, vol. 22, no. January, 2010.

[55] J. Leskovec and A. Krevl, “SNAP Datasets: Stanford large network dataset collection.” http://snap.stanford.edu/data, June 2014.

[56] S. R. . L. R. . W. S. . et al., “Social networks and the performance of individuals and groups,” *Academy of Management Journal*, vol. 44, pp. 316–325, April 2001.

[57] G. D. . R. NP, “Group decision-making under threat–the tycoon game,” *Academy of Management Journal*, vol. 28, pp. 613–627, 1985.

[58] M. P. J. Traud, Amanda L. and M. A. Porter, “Social structure of facebook networks,” *Physica A*, vol. 391, pp. 4165–4180, 2011.

[59] C. C. Christine Task, *Publicly Constrained Populations in Differential Privacy*, 2012. `http://www.cs.purdue.edu/homes/ctask/pdfs/PublicPopulationPoster.pdf`.

[60] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. 10008, 2008.

[61] H. Ibarra, “Paving an alternative route: Gender differences in managerial networks,” *Social Psychology Quarterly*, pp. 91–102, 1997.

[62] R. S. Burt, “The gender of social capital,” *Rationality and Society*, vol. 10, no. 1, pp. 5–46, 1998.

[63] M. Fenner, *Concepts and Applications of Inferential Statistics*, 1998-2015. `http://vassarstats.net/textbook/`.

[64] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics bulletin*, pp. 80–83, 1945.

[65] C. Whong, *FOILing NYCs Taxi Trip Data*, March 18, 2014 (accessed March 15, 2015). `http://chriswhong.com/open-data/foil_nyc_taxi/`.

[66] J. Trotter, *Public NYC TaxiCab Database Lets You See How Celebrities Tip*, October 23, 2014 (accessed March 15, 2015). `http://gawker.com/the-public-nyc-taxicab-database-that-accidentally-track-1646724546`.

[67] L. Fleisher and D. Macmillan, *How Sharp-Elbowed Uber Is Trying to Make Nice*, January 29, 2015 (accessed March 25, 2015). `http://www.wsj.com/articles/hard-driving-uber-gives-compromise-a-try-1422588782?tesla=y`.

[68] B. Fischer, *Uber Backs Down in Data Fight with NYC*, January 2015 2015 (access April 12, 2015). `m.bizjournals.com/newyork/blog/techflash/2015/01/uber-backs-down-in-data-fight-with-nyc.html?page=all&r=full`.

VITA

VITA

Christine Marie Task earned her B.S. in Mathematics at Ohio State University in 2003, and her M.S. in Computer Science at Indiana University in 2009. In September 2010, she was admitted to Purdue University, where she began research in privacy-preserving data-mining under the supervision of her advisor, Prof. Chris Clifton. While at Purdue, Christine was a graduate student member of the Center for Education and Research in Information Assurance and Security (CERIAS) and the Center for the Science of Information (CoSI). In May 2015 she earned the degree of Doctor of Philosophy in Computer Science. Her research interests include social network analysis, differential privacy, and general applications of abstract formal mathematics to concrete informal reality.